VERITAS™

VERITAS NetBackup™ 5.1 for Microsoft SQL Server

System Administrator's Guide

for Windows

N123448

Disclaimer

The information contained in this publication is subject to change without notice. VERITAS Software Corporation makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. VERITAS Software Corporation shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this manual.

VERITAS Legal Notice

VERITAS Software Corporation
350 Ellis Street
Mountain View, CA 94043
USA
Phone 650–527–8000 Fax 650–527–2901
www.veritas.com

Third-Party Copyrights

Contents

Preface xi

What Is In This Manual? xii

Getting Help xiii

NetBackup Manuals xiv

Glossary xvi

Accessibility Features xvi

Conventions xvi

Chapter 1. Introduction 1

Features 2

NetBackup Operations 2

Graphical User Interface 2

Chapter 2. Installation Requirements and Registering a License Key 5

NetBackup for SQL Server Platform Support 6

System Configuration Requirements 7

Installation Prerequisites 7

Registering NetBackup for SQL Server 7

Chapter 3. Configuration 9

Verifying Installed Components 10

Configuration Using the NetBackup Administration Console for Windows 12

Configuring Media Using Media and Device Management 13

Setting the Maximum Jobs Per Client Property 14

Configuring a NetBackup Policy 15
Adding New Policies 16
Description of Attributes 18
Adding New Schedules 19
Schedule Properties 21
Types of Backups 23
Specifying the List of Backup Selections 23
Adding Clients to a Policy 24
Configuration Using the NetBackup Administration Console for UNIX 26
Configuring Media Using Media and Device Management 27
Setting the Maximum Jobs Per Client Property 28
Configuring a NetBackup Policy 29
Adding New Policies 29
Description of Attributes 32
Adding New Schedules 32
Schedule Properties 35
Types of Backups 37
Specifying the List of Backup Selections 37
Adding Clients to a Policy 39
Testing NetBackup for SQL Server Configuration Settings 41
NetBackup Administration Console for Windows 41
NetBackup Administration Console for UNIX 42
Configuring for Multi-Stream Operations 45
Configuring for Non-Multiplexed Backup Stripes 45
Configuring for Multiplexed Backup Stripes 45
SQL Server Configuration 46
DBMS (Database Management System) Privileges 46
Setting the SQL Server Login for Scheduled Operations 46

Authorizing Scheduled Operations (for Sites with SQL Server Security Restrictions) 47
How to Use Standard SQL Server Security 49
Protecting Files and Filegroups 50
Recovery Considerations for Files and Filegroups 50
Recovery Factors 51
Transaction Logs 51
Recovery Strategies 51
Backing up the Transaction Log 52
Database Differential Backup 52
File and Filegroup Backups 53
Database Recovery 53
Staging Recovery 54
Backing Up SQL Server in an Environment with Log Shipping 54
Performance Factors 55
Buffer Space Parameters 55
MAXTRANSFERSIZE 55
BLOCKSIZE 55
NUMBER OF BUFFERS PER STRIPE 56
Backup Stripes 56
Shared Memory Usage 56
Alternate Buffer Method 57

Chapter 4. Using NetBackup for SQL Server 59
Using dbbackex 60
Using Client-based Schedulers with dbbackex 61
Using the NetBackup Database Client Graphical User Interface 62
Starting the NetBackup Database Client Graphical User Interface for the First Time 62
Setting the Database Login Parameters 63
NetBackup Backup Operations 65

Backing up a Database 65
Backing up a Transaction Log 66
Backing Up a Database Filegroup 69
Backing Up a Database File 71
NetBackup Restore Operations 75
Browsing Backup Images 75
Restoring a Database Backup 76
Restoring a Database Differential Backup 77
Restoring a Filegroup Backup 78
Restoring a Filegroup Differential Backup 79
Restoring a File 80
Restoring a Transaction Log Image Without Staging a Full Recovery 82
Restoring a Transaction Log Image Within a Staged Recovery 84
Performing a Database Move 85
Performing a Partial Database Restore 89
Redirecting a Restore to a Different Host 95
Redirecting a Database to a Different Location on a Different Host 96
Creating a Move Template for Images Created with Pre-4.5 Versions of NetBackup 100
Selecting a Master Server 100
Using Batch Files 101
Overview of Batch Files 101
Text Format Used in Batch Files 101
Guidelines for Creating Batch Files 102
Notes 102
Keywords and Values Used in Batch Files 103
Creating a Backup Batch File 108
Backing Up All Databases or Excluding Databases 111
Sample Batch Files 112
Running NetBackup Batch Files 120

Browsing Objects on Remote SQL Server Installations 122
Starting Multi-streamed Operations 124
Using Multiple-Streams for Advanced Backup Methods 124
Using bplist to Retrieve SQL Server Backups 125

Chapter 5. Disaster Recovery 127
Preparing for Disaster Recovery of SQL Server 128
Disaster Recovery of SQL Server 128
Disaster Recovery of SQL Server Databases 130

Chapter 6. Using NetBackup for SQL Server with Clustering Solutions 133
Support for Microsoft Cluster Server Clusters 134
Support for VERITAS Cluster Server Clusters 134
Installing NetBackup for SQL Server in a Cluster 134
Installation Prerequisites 135
Configuring the NetBackup Server to Be Aware of Clustered SQL Server Instances 136
Performing a Backup on a Virtual SQL Server Instance 137
Performing a Restore on a Virtual SQL Server Instance 137
Backing Up SQL Server in a Cluster with a Multi-Interface Network Connection . . 138
Master Server Configuration 138
Adding Clients to the Policy 138
Adding Permissions That Allow for Browsing of Backups Across the Private Interface 139
Client Configuration 140
Specifying the Private Name of the Client 140
Modifying the Host File on Each SQL Server Host 141
Performing Backups in a Cluster with a Multi-Nic Connection 141
Performing Restores in a Cluster with a Multi-Nic Connection 143

Chapter 7. Using NetBackup for SQL Server with a Multi-Interface Network Connection 145

Master Server Configuration 146

Adding Clients to the Policy 146

Adding Permissions That Allow for Browsing of Backups Across the Private Interface 147

Client Configuration 147

Specifying the Private Name of the Client 147

Performing Backups with a Multi-Nic Connection 149

Performing Restores with a Multi-Nic Connection 150

Chapter 8. Performing Backups and Restores in a SAP Environment 153

Policy Configuration 154

Creating Batch Files to be Used in Automatic Backup Schedules 154

Monitoring Backups 156

Restoring the R/3 Database 156

Including Differential Backups in a Restore Operation 156

Restoring the R/3 Database after a Disk Crash 157

Restoring the Database and Transaction Log Backups 157

Chapter 9. Using NetBackup for SQL Server with Advanced Client 159

NetBackup Advanced Client for SQL Server Overview 160

Advanced Client Features 161

Supported Platforms 161

How SQL Server Operations Use Advanced Client 162

Selection of Backup Method 162

What is Backed Up 162

Performance Considerations 163

Performing SQL Server Snapshot Backups 163

Performing SQL Server Snapshot Restores 163

Grouped Backups 164
Requirements 165
Performing a Grouped Backup 166
Restoring a Database Backed Up in a Group 168
Using a Cloaked Database Snapshot to Impact How Differentials are Based . . . 169
Sample Backup Schedule Using Cloaked Backups 169
Creating a Batch File for a Cloaked Backup 171
Configuring a Snapshot Backup 172
Configuration Requirements 172
Configuration Steps 172
Configuring Advanced Client Policies for NetBackup for SQL Server 173
Configuring an Advanced Client Policy 173
Configuring a Policy for Instant Recovery 175

Chapter 10. Using NetBackup to Restore SQL Server from Backup Exec Images . . . 179

Requirements for Restoring Backup Exec Images 180
Limitations When Restoring Backup Exec Images 180
Restoring from Backup Exec Images 181
Restore Options 181
Restore Options for Redirected Restores 185
Specifying the Server, Client, and Policy Type 186
Restoring Backup Exec Database Backups 187
Restoring Backup Exec SQL Transaction Logs Backups Up to a Point in Time . . 189
Restoring Backup Exec SQL Transaction Logs Backups Up to a Named Transaction 191
Restoring Backup Exec Filegroup Backups 193
Restoring the SQL Master Database from a Backup Exec Image 197
Redirecting a Restore 199

Chapter 11. Troubleshooting 201
Progress Reports Created for NetBackup for SQL Server on the Client 202
NetBackup for SQL Server Logs 205
Log Descriptions 205
bphdb Log 205
dbclient Log 205
Changing the Debug Level 206
NetBackup Reports 208
Preventing Timeout Failures on Large Database Restores 208

Chapter 12. Reference 209
NetBackup for Microsoft SQL Server Graphical User Interface 210
Menu Bar 211
Actions Menu 211
Options Menu 212
View Menu 213
Help Menu 214
Dialogs 215
Set database login parameters 215
Backup Microsoft SQL Server Objects 217
Client Options 220
Backup History Options 222
Restore Microsoft SQL Server Objects 224
Start Batch File 230

Index 233

Preface

This guide describes how to install, configure and use the NetBackup for Microsoft® SQL Server extension on a Windows platform.

This document is the same as `NetBackup_AdminGuide_MS-SQL_Win.pdf` distributed with the NetBackup for SQL Server software.

This guide is intended for the:

- SQL Server system administrator responsible for configuring and using SQL Server.
- NetBackup system administrator responsible for configuring NetBackup.

A system administrator is a person with system administrator privileges and responsibilities.

This guide assumes:

- A basic understanding of Windows system administration.
- A working understanding of the NetBackup for Windows client software.
- A working understanding of the NetBackup server and client software.
- A familiarity with the information covered in the following NetBackup manuals:
  - NetBackup System Administrator's Guide for UNIX, Volumes I and II, or NetBackup System Administrator's Guide for Windows, Volumes I and II.
  - *NetBackup Troubleshooting Guide for UNIX and Windows.*
- A thorough understanding of the SQL Server administration.

What Is In This Manual?

This guide is organized as follows:.

Chapters in This Manual

Chapter	Description
"Introduction"	Provides an overview of NetBackup for SQL Server and lists some features. Read this chapter to become familiar with NetBackup.
"Installation Requirements and Registering a License Key"	Describes the system configuration requirements and the installed components of NetBackup for SQL Server.
"Configuration"	Explains how to configure NetBackup for SQL Server.
"Using NetBackup for SQL Server"	Explains how to use NetBackup for SQL Server for installing and backing up transaction logs and databases.
"Disaster Recovery"	Describes how to prepare for and perform a disaster recovery of an SQL Server.
"Using NetBackup for SQL Server with Clustering Solutions"	Describes how to use NetBackup for SQL Server in cluster environments.
"Using NetBackup for SQL Server with a Multi-Interface Network Connection"	Describes how to perform backups of SQL Server using a multi-nic connection.
"Performing Backups and Restores in a SAP Environment"	Contains information about how to use NetBackup for SQL Server in an SAP environment.
"Using NetBackup for SQL Server with Advanced Client"	Describes how to used the Advanced Client option with NetBackup for SQL Server to create snapshot images for backing up SQL Server.
"Using NetBackup to Restore SQL Server from Backup Exec Images"	Instructions for configuring and using NetBackup to restore SQL Server from Backup Exec images.
"Reference"	Describes the various dialog boxes, screens, and menus associated with the NetBackup Database Extension Graphical User Interface.

Getting Help

VERITAS offers you a variety of support options.

Accessing the VERITAS Technical Support Web Site

The VERITAS Support Web site allows you to:

- obtain updated information about NetBackup, including system requirements, supported platforms, and supported peripherals
- contact the VERITAS Technical Support staff and post questions to them
- get the latest patches, upgrades, and utilities
- view the NetBackup Frequently Asked Questions (FAQ) page
- search the knowledge base for answers to technical support questions
- receive automatic notice of product updates
- find out about NetBackup training
- read current white papers related to NetBackup

The address for the VERITAS Technical Support Web site follows:

- http://support.veritas.com

Subscribing to VERITAS Email Notification Service

Subscribe to the VERITAS Email notification service to be informed of software alerts, newly published documentation, Beta programs, and other services.

Go to http://support.veritas.com. Select a product and click “E-mail Notifications” on the right side of the page. Your customer profile ensures you receive the latest VERITAS technical information pertaining to your specific interests.

Accessing VERITAS Telephone Support

Telephone support for NetBackup is only available with a valid support contract. To contact VERITAS for technical support, dial the appropriate phone number listed on the Technical Support Guide included in the product box and have your product license information ready for quick navigation to the proper support group.

▼ **To locate the telephone support directory on the VERITAS web site**

1. Open http://support.veritas.com in your web browser.
2. Click the **Phone Support** icon. A page that contains VERITAS support numbers from around the world appears.

Accessing VERITAS E-mail Support

▼ **To contact support using E-mail on the VERITAS web site**

1. Open http://support.veritas.com in your web browser.
2. Click the **E-mail Support** icon. A brief electronic form will appear and prompt you to:
   - Select a language of your preference
   - Select a product and a platform
   - Associate your message to an existing technical support case
   - Provide additional contact and product information, and your message
3. Click **Send Message**.

Contacting VERITAS Licensing

For license information call 1-800-634-4747 option 3, fax 1-650-527-0952, or e-mail amercustomercare@veritas.com.

NetBackup Manuals

The following manuals, along with the online help, are included in the NetBackup documentation set. For a more detailed listing of NetBackup documents, refer to *NetBackup Release Notes for UNIX and Windows*.

If you have a UNIX server, refer to these documents:

- *VERITAS NetBackup System Administrator's Guide for UNIX, Volume I*

  NetBackup_AdminGuideI_UNIXServer.pdf

  Explains how to configure and manage NetBackup on a UNIX server, including managing storage units, backup policies, catalogs and host properties.

- *VERITAS NetBackup System Administrator's Guide for UNIX, Volume II*

  NetBackup_AdminGuideII_UNIXServer.pdf

  Explains additional NetBackup features such as notify scripts, enhanced authorization and authentication, and role-based security. The guide also discusses using NetBackup with AFS, Intelligent Disaster Recovery (IDR), and the BE Tape Reader.

- *NetBackup Media Manager System Administrator's Guide for UNIX*

  Explains how to configure and manage the storage devices and media on UNIX NetBackup servers. Media Manager is part of NetBackup.

- *VERITAS NetBackup Troubleshooting Guide for UNIX and Windows*

  Provides troubleshooting information for UNIX- and Windows-based NetBackup products, including Media Manager.

If you have a Windows server, refer to these documents:

- *VERITAS NetBackup System Administrator's Guide for Windows, Volume I*

  NetBackup_AdminGuideI_WinServer.pdf

  Explains how to configure and manage NetBackup on a Windows server, including managing storage units, backup policies, catalogs and host properties.

- *VERITAS NetBackup System Administrator's Guide for Windows, Volume II*

  NetBackup_AdminGuideII_WinServer.pdf

  Explains additional NetBackup features such as notify scripts, enhanced authorization and authentication, and role-based security. The guide also discusses using NetBackup with AFS, Intelligent Disaster Recovery (IDR), and the BE Tape Reader.

- *NetBackup Media Manager System Administrator's Guide for Windows*

  Explains how to configure and manage the storage devices and media on Windows NetBackup servers. Media Manager is part of NetBackup.

- *VERITAS NetBackup Troubleshooting Guide for UNIX and Windows*

  Provides troubleshooting information for UNIX- and Windows-based NetBackup products, including Media Manager.

Glossary

If you encounter unfamiliar terminology, consult the NetBackup online glossary. The glossary contains terms and definitions for NetBackup and all additional NetBackup options and agents.

The NetBackup online glossary is included in the NetBackup help file.

▼ **To access the NetBackup online glossary**

1. In the NetBackup Administration Console, click **Help** > **Help Topics**.
2. Click the **Contents** tab.
3. Click **Glossary of NetBackup Terms**.

Use the scroll function to navigate through the glossary.

Accessibility Features

NetBackup contains features that make the user interface easier to use by people who are visually impaired and by people who have limited dexterity. Accessibility features include:

- Support for assistive technologies such as screen readers and voice input (Windows servers only)
- Support for keyboard (mouseless) navigation using accelerator keys and mnemonic keys

For more information, see the *NetBackup System Administrator's Guide for Windows, Volume I* or the *NetBackup System Administrator's Guide for UNIX, Volume I*.

Conventions

The following conventions apply throughout the documentation set.

Product-Specific Conventions

The following term is used in the NetBackup 5.1 documentation to increase readability while maintaining technical accuracy.

- Microsoft Windows, Windows

Terms used to describe a specific product or operating system developed by Microsoft, Inc. Some examples you may encounter in NetBackup documentation are, Windows servers, Windows 2000, Windows Server 2003, Windows clients, Windows platforms, or Windows GUI.

When Windows or Windows servers is used in the documentation, it refers to all of the currently supported Windows operating systems. When a specific Windows product is identified in the documentation, only that particular product is valid in that instance.

For a complete list of Windows operating systems and platforms that NetBackup supports, refer to the *NetBackup Release Notes for UNIX and Windows* or go to the VERITAS support web site at `http://www.support.veritas.com`.

Typographical Conventions

Here are the typographical conventions used throughout the manuals:

Conventions

Convention	Description
GUI Font	Used to depict graphical user interface (GUI) objects, such as fields, listboxes, menu commands, and so on. For example: Enter your password in the **Password** field.
Italics	Used for placeholder text, book titles, new terms, or emphasis. Replace placeholder text with your specific text. For example: Replace *filename* with the name of your file. Do *not* use file names that contain spaces. This font is also used to highlight NetBackup server-specific or operating system-specific differences. For example: *This step is only applicable for NetBackup Enterprise Server.*
`Code`	Used to show what commands you need to type, to identify pathnames where files are located, and to distinguish system or application text that is displayed to you or that is part of a code example.
Key+Key	Used to show that you must hold down the first key while pressing the second key. For example: Ctrl+S means hold down the Ctrl key while you press S.

You should use the appropriate conventions for your platform. For example, when specifying a path, use backslashes on Microsoft Windows and slashes on UNIX. Significant differences between the platforms are noted in the text.

Tips, notes, and cautions are used to emphasize information. The following samples describe when each is used.

Tip Used for nice-to-know information, like a shortcut.

Note Used for important information that you should know, but that shouldn't cause any damage to your data or your system if you choose to ignore it.

Caution Used for information that will prevent a problem. Ignore a caution at your own risk.

Command Usage

The following conventions are frequently used in the synopsis of command usage.

brackets [ ]

The enclosed command line component is optional.

Vertical bar or pipe (|)

Separates optional arguments from which the user can choose. For example, when a command has the following format:

`command arg1|arg2`

In this example, the user can use either the *arg1* or *arg2* variable.

Navigating Multiple Menu Levels

When navigating multiple menu levels, a greater-than sign (>) is used to indicate a continued action.

The following example shows how the > is used to condense a series of menu selections into one step:

- Select **Start** > **Programs** > **VERITAS NetBackup** > **NetBackup Administration Console**.

The corresponding actions could be described in more steps as follows:

1. Click **Start** in the task bar.
2. Move your cursor to **Programs**.
3. Move your cursor to the right and highlight **VERITAS NetBackup**.
4. Move your cursor to the right. First highlight and then click **NetBackup Administration Console**.

Terms Used in this Document

Microsoft SQL Server will be referred to as SQL Server. In this manual, NetBackup for Microsoft SQL Server will be referred to as NetBackup for SQL Server.

Introduction 1

NetBackup for SQL Server extends the capabilities of NetBackup for Windows to include backing up and restoring SQL Server databases. These capabilities are provided for a Windows client using either a UNIX or Windows NetBackup master server.

NetBackup for SQL Server includes a client-based graphical user interface (GUI) program to perform various activities on SQL Server. These activities include:

- Setting options for NetBackup for SQL Server operations.
- Backing up, restoring and copying databases and database components, which include transaction logs, differentials, files and filegroups.
- Starting NetBackup for SQL Server operations from batch files which you have created.
- Monitoring NetBackup for SQL Server operations.

Features

This section describes the main features of NetBackup for SQL Server.

NetBackup Operations

- Full integration with the NetBackup master server and Media Manager.
- Stream-based backup and restore of SQL Server objects to tape or disk using SQL Server's high-speed virtual device interface.
- Snapshot-based backup and restore of SQL Server objects using NetBackup Advanced Client methods.
- Backup and recovery of databases, differentials, files, filegroups, and transaction logs.
- Browse capability for SQL Server objects on the local and remote nodes.
- Support for redirection of SQL Server restores to different locations.
- Support for multiple SQL Server instances.
- Client operation monitoring through the NetBackup Client Job Monitor. Server monitoring is also available through the NetBackup master.
- Performance tuning through user control of backup stripes, transfer size, and buffer usage.
- Job launch is supported through the following options:
  - Immediate launch through the NetBackup Database Extension GUI
  - Scheduled launch through the NetBackup scheduler
  - Command line launch
- Support for Microsoft Cluster Server and Veritas Cluster Server.
- Recovery of Microsoft SQL images backed up with Backup Exec, through the Backup, Archive, and Restore (NetBackup Client) interface.

Graphical User Interface

- GUI capability for browsing:
  - SQL Server databases
  - SQL Server database files and filegroups
- GUI assistance for staging a complete database recovery from backup images created for databases, filegroups, files, database differentials, and transaction logs.

- GUI assistance for restoring database objects backed up on one SQL Server client to another SQL Server client.
- GUI assistance for creating and saving a backup script as an alternative to performing an immediate job launch.
- Online help provided through the NetBackup Database Extension GUI.

Installation Requirements and Registering a License Key 2

This chapter describes the installation prerequisites for NetBackup for SQL Server and how to register a license key for this agent.

For information on installing NetBackup for SQL Server in a cluster, refer to "Using NetBackup for SQL Server with Clustering Solutions" on page 133.

NetBackup for SQL Server Platform Support

The VERITAS support web site now contains the most current platform support information for NetBackup database agents. You can use the following procedure to locate this information.

▼ To locate the latest Database Agent information on the VERITAS support web site

1. Go to the VERITAS support web page: http://www.support.veritas.com.
2. From the Select Product Family list, choose **NetBackup Products**. The Select Product window refreshes showing a list of NetBackup products.
3. Select one of the NetBackup products. The page refreshes and displays a search window with four tabs:
   - Documents
   - Downloads
   - Software Alerts
   - Compatibility
4. Select **5.1** in the **Product Version** drop-down field.
5. Select a Product Platform.
6. Click the **Compatibility** tab. The page will refresh showing documents that contain compatibility information.
7. Click on the title of the document link for NetBackup Database Agent Compatibility List. A document appears that contains a downloadable file that enables you to view the supported database spreadsheet for this release.

System Configuration Requirements

To use NetBackup for SQL Server, you need:

- Windows NT 4.0 with Service Pack 3 or higher, or Windows 2000 or later
- NetBackup for UNIX or NetBackup for Windows
- SQL Server 7.0 with Service Pack 1 or higher, or SQL Server 2000 with Service Pack 3 or higher
- In order to use NetBackup advanced client methods (snapshots) for backup and restore, Windows 2000 or 2003 is required.

Note SQL Server 6.5 objects that were backed up with earlier versions of this product can still be restored. However, the current release does not support backing up SQL Server 6.5.

Installation Prerequisites

Before enabling NetBackup for SQL Server, be sure to complete the following procedures.

- Install NetBackup server software on the server.
- Install the NetBackup client software on the client where you will be backing up the databases. This step also installs NetBackup for SQL Server.

  See the *NetBackup Installation Guide for Windows* for installation instructions on Windows clients.

Registering NetBackup for SQL Server

NetBackup for SQL Server is installed with the server and client software. To use this agent you need to register a valid license key for it on the master or media server.

▼ **To register a license key**

❖ Open the NetBackup Administration Console on the master or media server and choose **Help** > **License Keys**.

Refer to the *NetBackup System Administrator's Guide for Windows, Volume I* for full details on adding license keys.

Configuration 3

This chapter describes how to configure NetBackup for SQL Server. The following sections in this chapter describe each of these steps in detail.

- Verifying Installed Components
- Configuration Using the NetBackup Administration Console for Windows
- Configuring Media Using Media and Device Management
- Setting the Maximum Jobs Per Client Property
- Adding New Policies
- Configuring for Multi-Stream Operations
- SQL Server Configuration

This chapter also contains "Testing NetBackup for SQL Server Configuration Settings" on page 41 and "Performance Factors" on page 55.

Verifying Installed Components

Note *install_path* refers to the directory where you installed the NetBackup software. By default, this directory is `C:\Program Files\VERITAS\`.

The NetBackup SQL Server agent is installed automatically when you perform a NetBackup server or client installation. During the installation, the following actions for the NetBackup for SQL Server Agent were taken.

- An icon was added to the NetBackup Program Group

  Veritas NetBackup > **NetBackup Agents** > **NetBackup MS SQL Client**

- The following new directories were created, if they did not already exist on your Windows host:

  install_path`\NetBackup\DbExt\Mssql`

  The default directory for batch files and miscellaneous items pertaining to installation.

  install_path`\NetBackup\logs\user_ops\Mssql\jobs`

  The directory containing NetBackup master server logging files for each operation that you perform.

  install_path`\NetBackup\logs\user_ops\Mssql\logs`

  The directory containing NetBackup client logging files for each operation that you perform.

- Two sample *batch* files were placed in *install_path*`\NetBackup\DbExt\Mssql`:

  `bkup.bch`

  `rest.bch`

  Batch files are described in "Using Batch Files" on page 101.

- The following lists binaries created for NetBackup for SQL Server. The path for these components is *install_path*`\NetBackup\bin\`.

`DBbackmain.dll`	NetBackup client DLL which backs up and restores SQL Server databases.
`dbbackup.exe`	Graphical user interface to `dbbackmain`.

`dbbackex.exe`	A command-line interface to `dbbackmain`. It is also called internally by the NetBackup client (through `bphdb`), and by NetBackup for SQL Server for performing NetBackup scheduling operations.
`tdbackmain.dll`	Additional NetBackup client functions.
`xbsa.dll`	Common NetBackup database client functions

Configuration Using the NetBackup Administration Console for Windows

Although the database agent is installed on the NetBackup client, some configuration procedures are performed using the NetBackup Administration Console on the server.

These procedures include:

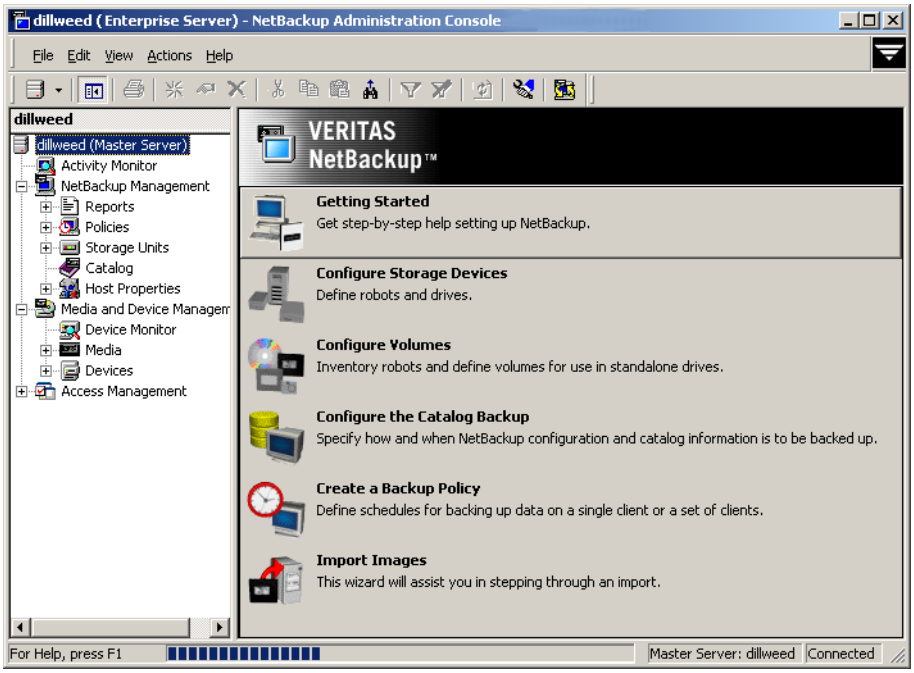
- Configuring media using Media and Device Management
- Setting the **Maximum Jobs per Client** property
- Configuring a NetBackup policy
- Testing NetBackup for SQL Server configuration settings

See the next section for instructions on starting the NetBackup Administration Console.

▼ To launch the NetBackup Administration Console for Windows

1. Log on to the server as administrator.
2. Choose **Start** > **Programs > VERITAS NetBackup** > **NetBackup Administration Console**.

The NetBackup Administration Console is displayed.



Configuring Media Using Media and Device Management

Use the Media and Device Management (Media Manager) to configure media for a NetBackup for SQL Server configuration. Refer to the *NetBackup Media Manager System Administrator's Guide* for more information.

The number of volumes required will depend on the devices used, the size of the SQL Server databases that you are backing up, the size of your backups, and the frequency of backups.

Setting the Maximum Jobs Per Client Property

The **Maximum jobs per client** attribute value is figured with the following formula.

Maximum Jobs per Client = *Number of Streams* x *Number of Policies*

Where:

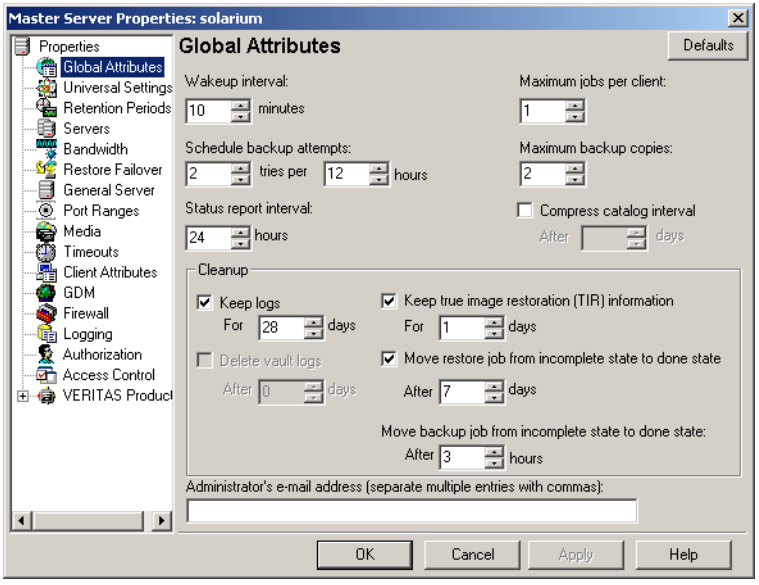
- *Number of Streams* is the number of backup streams between the database server and NetBackup. If striping is not used, then each separate stream starts a new backup job on the client. If striping is used, then each new job uses one stream per stripe.
- *Number of Policies* is the number of policies that may back up this client at the same time. This number can be greater than one. For example, a client may be in two policies in order to back up two different databases. These backup windows may overlap.

▼ To set the Maximum jobs per client property using the Windows console

1. In the left pane of the NetBackup Administration Console, expand **Host Properties**.
2. Select **Master Server**.
3. In the right pane, double-click on the server icon.

   The Master Server Properties dialog is displayed.

4. In the Master Server Properties dialog, click **Global Attributes**.



The default value is 1 for **Maximum jobs per client**.

5. Change the **Maximum jobs per client** value to 99.

   If you prefer to enter a lower value, calculate the value equal to the maximum number of backups allowed per client using the formula provided above.

Configuring a NetBackup Policy

A NetBackup policy defines the backup criteria for a specific group of one or more clients. These criteria include:

- storage unit and media to use
- backup schedules
- clients to be backed up
- batch file files to be executed on the clients

To use NetBackup for SQL Server, at least one MS-SQL-Server policy with the appropriate schedules needs to be defined. A configuration can have a single policy that includes all clients or there can be many policies, some of which include only one client.

Most requirements for MS-SQL-Server policies are the same as for file system backups. In addition to the attributes described here, there are other attributes for a policy to consider. Refer to the *NetBackup System Administrator's Guide for Windows, Volume I* or *NetBackup System Administrator's Guide for UNIX, Volume I* for detailed configuration instructions and information on all the attributes available.

Adding New Policies

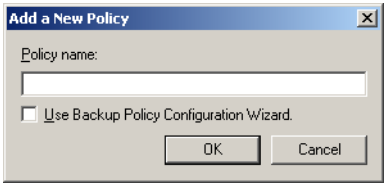
Use this procedure when configuring a policy from a Windows server or from a NetBackup Remote Administration Console host.

▼ To add a new policy

Note If you are going to perform multi-streamed backups and restores, see "Configuring for Multi-Stream Operations" on page 45

1. Log on to the server as administrator.
2. Start the NetBackup Administration Console.
3. If your site has more than one master server, choose the one where you want to add the policy.
4. In the left pane, right-click **Policies** and choose **New Policy**.

   The Add a New Policy dialog is displayed.



   a. In the **Policy name** box, type a unique name for the new policy.

   b. Choose whether to use the wizard for configuring the policy. The wizard guides you through most of the setup process and simplifies it by automatically choosing default values that are good for most configurations. If necessary, you can change the defaults later by editing the policy. You will need to manually add entries to the Backup Selections list after completing the wizard.

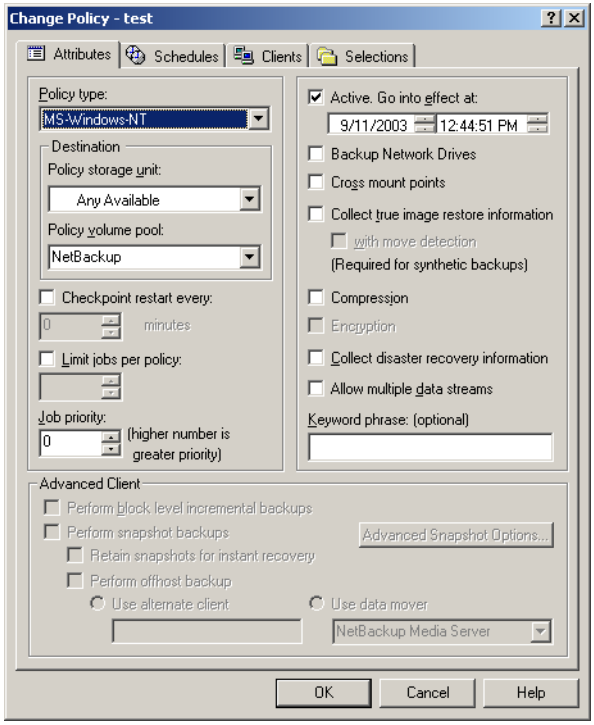
      - To use the wizard, select the **Use Backup Policy Configuration Wizard** box and click **OK**. The wizard starts and you create the policy by following the prompts. When prompted, select the MS-SQL-Server policy type.

Note The MS-SQL-Server policy type will not appear in the drop-down list unless your server has a license key for NetBackup for SQL Server.

- If you require more control over the settings than the wizard provides, then do not select the **Use Backup Policy Configuration Wizard** box and proceed to step 5.

5. Click **OK**.

   A dialog is displayed in which you can specify the general attributes for the policy.



6. From the **Policy Type** box, select the MS-SQL-Server policy type.

7. Complete the entries on the **Attributes** tab as explained in "Description of Attributes."

8. Add other policy information:
   - To add schedules, see "Adding New Schedules."
   - To add clients, see "Adding Clients to a Policy."
   - To add batch files to the Backup Selections list, see "Specifying the List of Backup Selections."
9. Click **OK**. The new policy is created.

Description of Attributes

With a few exceptions, NetBackup manages a database backup like a file system backup. Policy attributes that are different for SQL Server backups are explained below.

Your other policy attributes will vary according to your specific backup strategy and system configuration. Consult the *NetBackup System Administrator's Guide for Windows, Volume I* or *NetBackup System Administrator's Guide for UNIX, Volume I* for detailed explanations of the policy attributes.

Description of Policy Attributes

Attribute	Description
Policy type	Determines the type of clients that can be in the policy and in some cases the types of backups that can be performed on those clients. To use NetBackup for SQL Server, you must have defined at least one MS-SQL-Server policy.
Keyword phrase to associate with the backup	For NetBackup for SQL Server, the keyword phrase entry is ignored.
Allow multiple data streams	Specifies that, depending on directives in the list of backup selections, NetBackup can divide automatic backups for each client into multiple jobs, with each job backing up only a part of the list of backup selections. The jobs are in separate data streams and can occur concurrently. The number of available storage units, multiplex settings, and the maximum jobs parameters determines the total number of streams and how many can run concurrently.
Advanced Client	See the "Using NetBackup for SQL Server with Advanced Client" chapter for information on configuring policies for snapshot methods.

Adding New Schedules

Each policy has its own set of schedules. These schedules control initiation of automatic backups and also specify when user operations can be initiated.

A SQL Server backup requires an Application Backup schedule, which is created automatically when you create a SQL Server policy. The Application Backup schedule is used to initiate and manage the backup operation. You also need one or more Automatic Backup schedules, if you plan to have NetBackup perform automatic, scheduled backups.

The following procedures explain how to configure the required schedule types, and how to add new schedules. The procedures start on the Policy dialog. To see the Policy dialog, double-click the policy name in the Policies list of the NetBackup Administration Console.

▼ To configure an Application Backup schedule

1. On the Policy dialog, click the **Schedules** tab.
2. Double-click on the schedule named **Default-Application-Backup**.

   A dialog is displayed.
3. Specify the other properties for the schedule as explained in "Schedule Properties."

   The backup window for an Application Backup schedule must encompass the time period during which all NetBackup jobs, scheduled and unscheduled, will occur. This is necessary because the Application Backup schedule starts processes that are required for all NetBackup for SQL Server backups, including those started automatically.

   For example, assume that you:

   - expect users to perform NetBackup operations during business hours, 0800 to 1300.
   - configured automatic backups to start between 1800 and 2200.

   The Application Backup schedule must have a start time of 0800 and a duration of 14 hours.

Example Settings for an Application Backup schedule

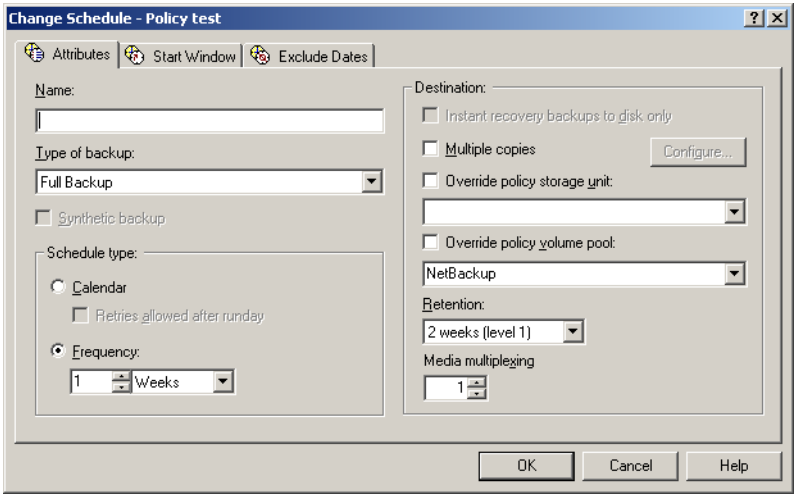
Type of Backup	Schedule settings	Description	Settings
Application Backup	Retention	The length of time backup images are stored.	2 weeks
	Backup Window	The time during which a NetBackup operation can be performed.	Sunday through Saturday 00:00:01 - 23:59:59

Tip Set the time period for the Application Backup schedule for 24 hours per day, seven days per week. This ensures that your NetBackup for SQL Server operations are never locked out due to the Application Backup schedule.

▼ To configure an automatic backup schedule

1. On the Policy dialog, click the **Schedules** tab.
2. Click **New**.

   A dialog is displayed. The title bar shows the name of the policy to which you are adding the schedules.



3. Specify a unique name for the schedule.
4. Select the **Type of backup**.

   For information on the types of backups available for this policy, see "Types of Backups."

Tip If you put multiple batch files in the same MS-SQL-Server policy, they will execute during each Automatic Backup session for that MS-SQL-Server policy. If you have a variety of SQL Server backup operations that you wish to have executed on different schedules, then consider creating multiple MS-SQL-Server policies with differing Automatic Backup schedules; and then assign each batch file to the policy that uses the desired Automatic Backup schedule.

Refer to the following table for example settings for an Automatic Backup schedule.

Example Settings for an Automatic Backup Schedule

Type of Backup	Schedule settings	Description	Settings
Automatic Backup	Retention	The length of time to store the record of a backup.	2 weeks
	Frequency	Frequency determines how often a backup should be performed	every week
	Backup Window	The time during which a NetBackup operation can be performed.	Sunday, 18:00:00 - 22:00:00

5. Specify the other properties for the schedule as explained in "Schedule Properties."
6. Click **OK**.

Schedule Properties

Some of the schedule properties have a different meaning for database backups than for a regular file system backup. These properties are explained in the following table.

Other schedule properties vary according to your specific backup strategy and system configuration. Consult the *NetBackup System Administrator's Guide for Windows, Volume I* or *NetBackup System Administrator's Guide for UNIX, Volume I* for detailed explanations of the schedule properties.

Description of Schedule Properties

Property	Description
Type of backup	Specifies the type of backup that this schedule will control. The selection list shows only the backup types that apply to the policy you are configuring. For more information see the following section "Types of Backups."
Frequency	This setting is used only for scheduled backups, and not for user-directed backups. **Frequency** specifies the period of time that will elapse until the next backup operation can begin on this schedule. For example, if the frequency is seven days and a successful backup occurs on Wednesday, the next full backup will not occur until the following Wednesday.

Description of Schedule Properties (continued)

Property	Description
Calendar	This setting is used only for scheduled backups, and not for user-directed backups. The **Calendar** option allows you to schedule backup operations based on specific dates, recurring week days, or recurring days of the month.
Retention	**Frequency based scheduling** The retention period for an Application Backup Policy schedule refers to the length of time that NetBackup keeps backup images. For example, if your database is backed up once every Sunday morning, you should select a retention period of at least 2 weeks.The retention period for an Automatic Backup schedule controls how long NetBackup keeps records of when scheduled backups have occurred. The NetBackup scheduler compares the latest record to the frequency to determine whether a backup is due. This means that if you set the retention period to expire the record too early, the scheduled backup frequency will be unpredictable. However, if you set the retention period to be longer than necessary, the NetBackup catalog will accumulate unnecessary records. *Therefore, set a retention period that is longer than the frequency setting for the schedule.* For example, if the frequency setting is set to one week, set the retention period to be more than one week. **Calendar based scheduling** The retention period for an Application Backup Policy schedule refers to the length of time that NetBackup keeps backup images. The retention period for an Automatic Backup schedule controls how long NetBackup keeps records of when scheduled backups have occurred. However, this setting is not significant for calendar based scheduling.
Multiple copies	If you are licensed for the Inline Tape Copy feature and wish to specify multiple copies for your MS-SQL-Server policy, configure **Multiple copies** on the Application Backup schedule.

Types of Backups

Description of Backup Types for SQL Server

Application Backup	The Application Backup schedule enables user-controlled NetBackup operations performed on the client. At least one Application Backup schedule type must be configured for each MS-SQL-Server policy. The Default-Application-Backup schedule is automatically configured as an Application Backup schedule.
Automatic Backup	An Automatic Backup schedule specifies the dates and times when NetBackup will automatically start backups by running the batch files in the order that they appear in the file list. If there is more than one client in the MS-SQL-Server policy, the batch files are executed on each client.

Specifying the List of Backup Selections

The Backup Selections list in a database policy has a different meaning than for non-database policies. For example, in a Windows policy, the list contains files and folders to be backed up. In a database policy, you can specify batch files to be run. NetBackup for SQL Server scripts are called batch files and are identified by the .bch extension. Batch files describe the backup and restore operations you wish to start through manual or scheduled operations initiated from the NetBackup server. These files reside on the client and direct the operation of NetBackup for SQL Server and SQL Server. Refer to "Overview of Batch Files" on page 101 for details on how to create a batch file.

Add batch files only if you are setting up a policy for automatic scheduling. All batch files listed in the Backup Selections list will be executed for the Automatic Backup schedules, as specified under the **Schedules** tab, and manual backups. NetBackup will start backups according to the order that batch files appear in the Backup Selections list.

The following procedure starts in the Policy dialog. To see the Policy dialog, double-click the policy name in the Policies list in the NetBackup Administration Console.

▼ To add batch files to the Backup Selections list

1. In the Policy dialog, click the **Backup Selections** tab.
2. Click **New**.

3. Enter the name of the selection.

   - Type the name of the batch file.

     Specify the full pathname for the batch file. For example, a path name for a batch file that will back up a database may be:

     install_path\NetBackup\DbExt\Mssql\bkup.bch

Note *install_path* refers to the directory where you installed the NetBackup software. By default, this directory is C:\Program Files\Veritas\.

     You can also specify just the file name if the batch file resides in the *install_path*\NetBackup\Dbext\Mssql\ directory.

     OR

   - Click the **Remote Folder** button.

     The Browse window is displayed. Navigate to the batch file and select the file. Click **OK**.

4. Click **OK**.

Adding Clients to a Policy

The client list is the list of hosts on which your batch files are run during an automatic backup. A NetBackup client must be in at least one policy but can be in more than one.

This procedure starts in the Policy dialog. To see the Policy dialog, double-click the policy name in the Policies list in the NetBackup Administration Console.

Requirements for clients

The following software must be installed on the client:

- SQL Server
- NetBackup client or server
- NetBackup for SQL Server
- the backup or restore script(s)

▼ To add clients to a policy

Note NetBackup for SQL Server must be installed on each client. Also, each batch file must be present on each client.

1. In the Policy dialog, click the **Clients** tab.
2. Click **New**.
3. Enter the name of the client you want to add.
   - Type the name into the client list and click **OK**.

     If NetBackup cannot detect the hardware and operating system, a dialog will be displayed so you can specify this information.

     OR

   - Click the **Browse for Computer** button to choose the client from the network:

   If SQL Server is installed in a cluster, specify the virtual SQL Server name as the client name.
4. To add another client, repeat step 2 and step 3. If this is the last client, click **OK** to close the dialog.

Configuration Using the NetBackup Administration Console for UNIX

Although the database agent is installed on the NetBackup client, some configuration procedures are performed using the NetBackup Administration Console on the server.

These procedures include:

- Configuring media using Media and Device Management
- Setting the **Maximum Jobs per Client** property
- Configuring a NetBackup policy
- Testing NetBackup for SQL Server configuration settings

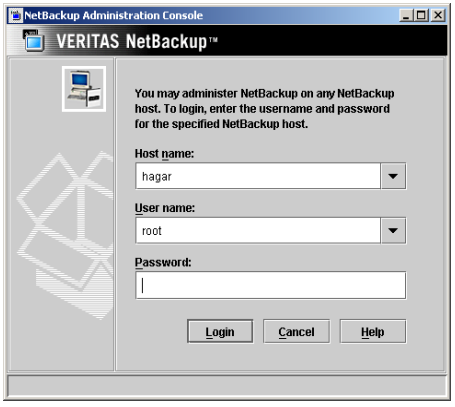
See the next section for instructions on starting the NetBackup Administration Console.

▼ To launch the NetBackup Administration Console for UNIX

1. Log onto the UNIX server as root.
2. Start the NetBackup Administration Console by executing:

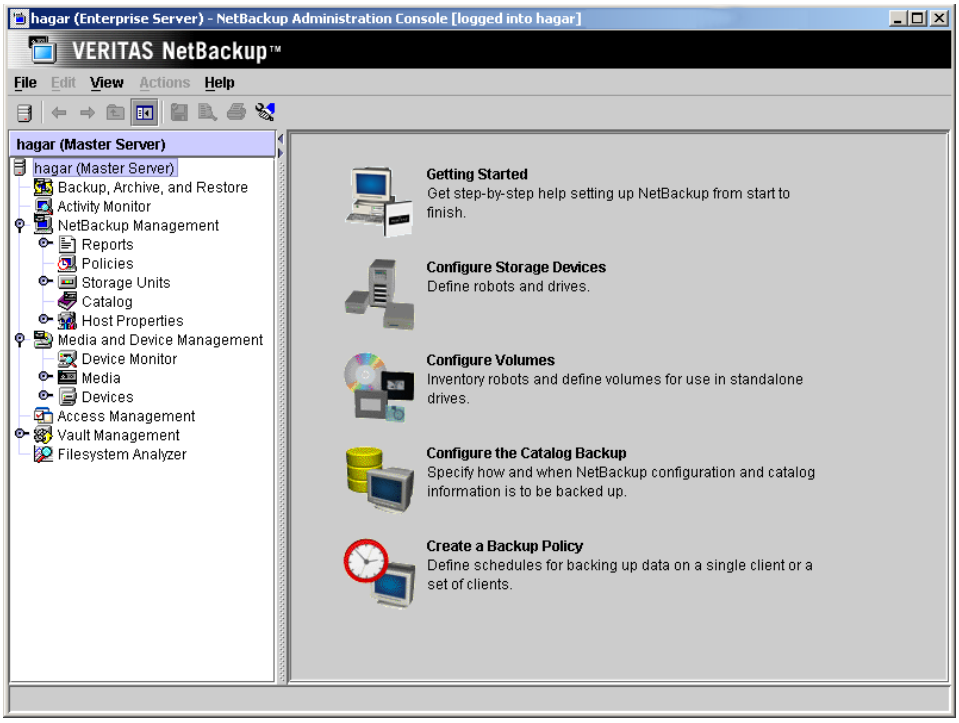
   install_path`/netbackup/bin/jnbSA &`

   The login dialog is displayed.



3. Type the name of the master server where you initially want to manage NetBackup. You can specify any NetBackup master server. Indicate the User and Password.

4. Click **Login**. The NetBackup Administration Console is displayed.



Configuring Media Using Media and Device Management

Use the Media and Device Management (Media Manager) to configure media for a NetBackup for SQL Server configuration. Refer to the *NetBackup Media Manager System Administrator's Guide* for more information.

The number of volumes required will depend on the devices used, the size of the SQL Server databases that you are backing up, the size of your backups, and the frequency of backups.

Setting the Maximum Jobs Per Client Property

The **Maximum jobs per client** attribute value is figured with the following formula.

Maximum Jobs per Client = *Number of Streams* x *Number of Policies*

Where:

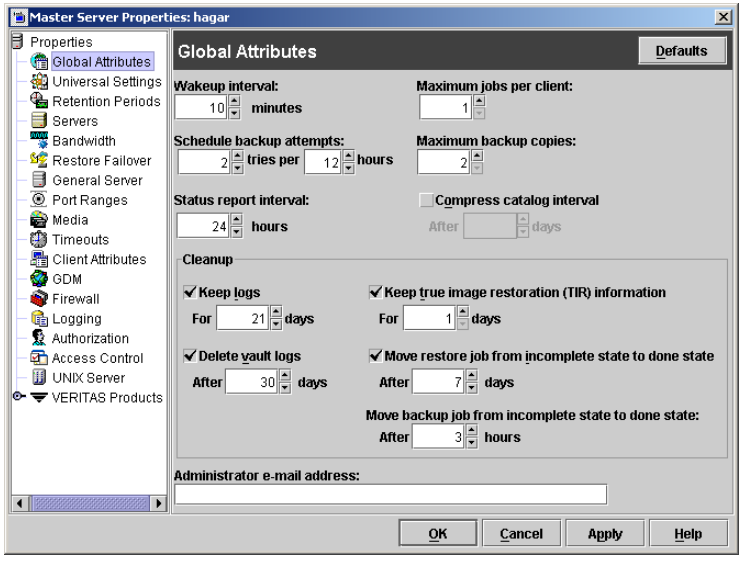
- *Number of Streams* is the number of backup streams between the database server and NetBackup. If striping is not used, then each separate stream starts a new backup job on the client. If striping is used, then each new job uses one stream per stripe.
- *Number of Policies* is the number of policies that may back up this client at the same time. This number can be greater than one. For example, a client may be in two policies in order to back up two different databases. These backup windows may overlap.

▼ To set the Maximum jobs per client property using the NetBackup Administration Console for UNIX

1. In the left pane of the NetBackup Administration Console, expand **Host Properties**. Select **Master Servers**.

2. In the right pane, double-click on the server icon. Click **Global Attributes**.

   The Master Server Properties dialog is displayed.



   The default value is 1 for **Maximum jobs per client**.

3. Change the **Maximum jobs per client** value to 99.

   If you prefer to enter a lower value, calculate the value equal to the maximum number of backups allowed per client using the formula provided above.

Configuring a NetBackup Policy

A NetBackup policy defines the backup criteria for a specific group of one or more clients. These criteria include:

- storage unit and media to use
- backup schedules
- clients to be backed up
- batch file files to be executed on the clients

To use NetBackup for SQL Server, at least one MS-SQL-Server policy with the appropriate schedules needs to be defined. A configuration can have a single policy that includes all clients or there can be many policies, some of which include only one client.

Most requirements for MS-SQL-Server policies are the same as for file system backups. In addition to the attributes described here, there are other attributes for a policy to consider. Refer to the *NetBackup System Administrator's Guide for Windows, Volume I* or *NetBackup System Administrator's Guide for UNIX, Volume I* for detailed configuration instructions and information on all the attributes available.

Adding New Policies

Use this procedure when configuring a policy from a UNIX server.

▼ **To add a new policy**

Note If you are going to perform multi-streamed backups and restores, see "Configuring for Multi-Stream Operations."

1. Log onto the server as root.
2. Start the NetBackup Administration Console.
3. If your site has more than one master server, choose the one to which you want to add the policy.

4. In the left pane, click on **Policies**. The right pane splits into an All Policies pane and a details pane.

5. In the All Policies pane, right-click on the Master Server, and click **New**.

   The Add a New Policy dialog is displayed.

   a. In the **Policy name** box, type a unique name for the new policy.

   b. Choose whether to use the wizard for configuring the policy. The wizard guides you through most of the setup process and simplifies it by automatically choosing default values that are good for most configurations. If necessary, you can change the defaults later by editing the policy. You will need to manually add entries to the Backup Selections list after completing the wizard.

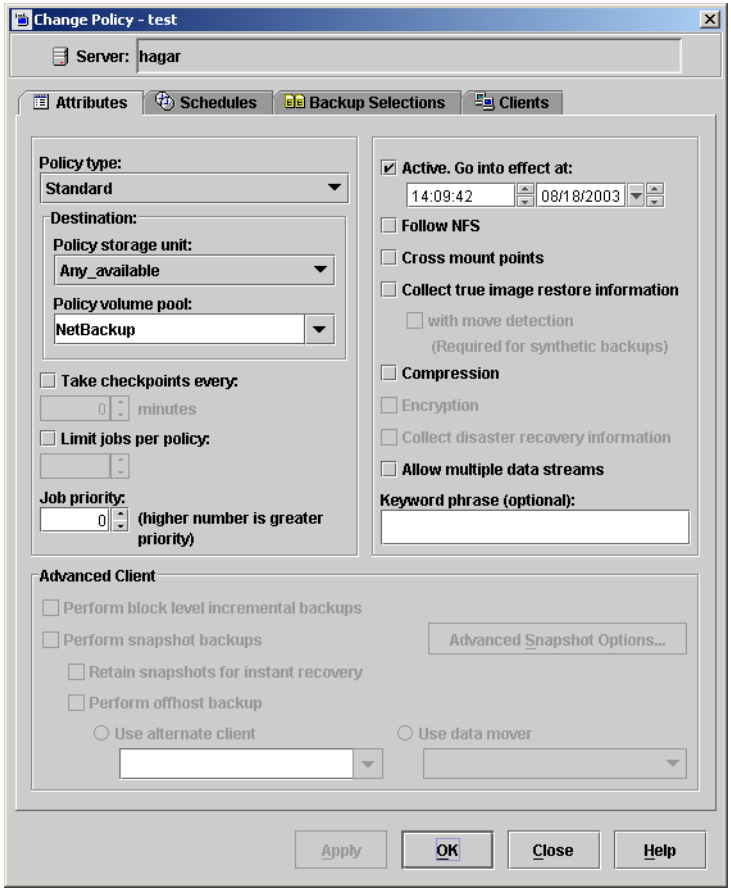
      - To use the wizard, select the **Use add policy wizard** box and click **OK**. The wizard starts and you create the policy by following the prompts. When prompted, select the MS-SQL-Server policy type.

Note The MS-SQL-Server policy type will not appear in the drop-down list unless your server has a license key for NetBackup for SQL Server.

      - If you require more control over the settings than the wizard provides, do not select the **Use add policy wizard** box and proceed to step 6.

6. Click **OK**.

A dialog is displayed in which you can specify the general attributes for the policy.



7. From the **Policy type** box, select the MS-SQL-Server policy type.

8. Complete the entries on the **Attributes** tab as explained in "Description of Attributes" and click **Apply** to save the attribute entries.

9. Add other policy information:
   - To add schedules, see "Adding New Schedules."
   - To add clients, see "Adding Clients to a Policy."
   - To add scripts, see "Specifying the List of Backup Selections."

10. Click **Apply**. The new policy is created.

Description of Attributes

With a few exceptions, NetBackup manages a database backup like a file system backup. Policy attributes that are different for SQL Server backups are explained below.

Your other policy attributes will vary according to your specific backup strategy and system configuration. Consult the *NetBackup System Administrator's Guide for Windows, Volume I* or *NetBackup System Administrator's Guide for UNIX, Volume I* for detailed explanations of the policy attributes.

Description of Policy Attributes

Attribute	Description
Policy type	Determines the type of clients that can be in the policy and in some cases the types of backups that can be performed on those clients. To use NetBackup for SQL Server, you must have defined at least one MS-SQL-Server policy.
Keyword phrase to associate with the backup	For NetBackup for SQL Server, the keyword phrase entry is ignored.
Allow multiple data streams	Specifies that, depending on directives in the list of backup selections, NetBackup can divide automatic backups for each client into multiple jobs, with each job backing up only a part of the list of backup selections. The jobs are in separate data streams and can occur concurrently. The number of available storage units, multiplex settings, and the maximum jobs parameters determines the total number of streams and how many can run concurrently.
Advanced Client	See the "Using NetBackup for SQL Server with Advanced Client" chapter for information on configuring policies for snapshot methods.

Adding New Schedules

Each policy has its own set of schedules. These schedules control initiation of automatic backups and also specify when user operations can be initiated.

A SQL Server backup requires at least two specific schedule types: an Application Backup schedule and an Automatic Backup schedule. You can also create additional schedules.

The following procedures explain how to configure the required schedule types, and how to add other new schedules. The procedures start from the Policy dialog. To see the Policy dialog, double click the policy in the policy list in the left pane of the NetBackup Administration Console.

▼ To configure an Application Backup schedule

1. In the Policy dialog, click the **Schedules** tab.
2. In the Schedule list, double-click on the schedule named **Default-Application-Backup**.

   A dialog is displayed. The title bar shows the name of the policy to which you are adding the schedule.
3. Specify the other properties for the schedule as explained in "Schedule Properties."

   The backup window for an Application Backup schedule must encompass the time period during which all NetBackup jobs, scheduled and unscheduled, will occur. This is necessary because the Application Backup schedule starts processes that are required for all NetBackup for SQL Server backups, including those started automatically.

   For example, assume that you:

   - expect users to perform NetBackup operations during business hours, 0800 to 1300.
   - configured automatic backups to start between 1800 and 2200.

   The Application Backup schedule must have a start time of 0800 and a duration of 14 hours.

Tip Set the time period for the Application Backup schedule for 24 hours per day, seven days per week. This ensures that your NetBackup for SQL Server operations are never locked out due to the Application Backup schedule.

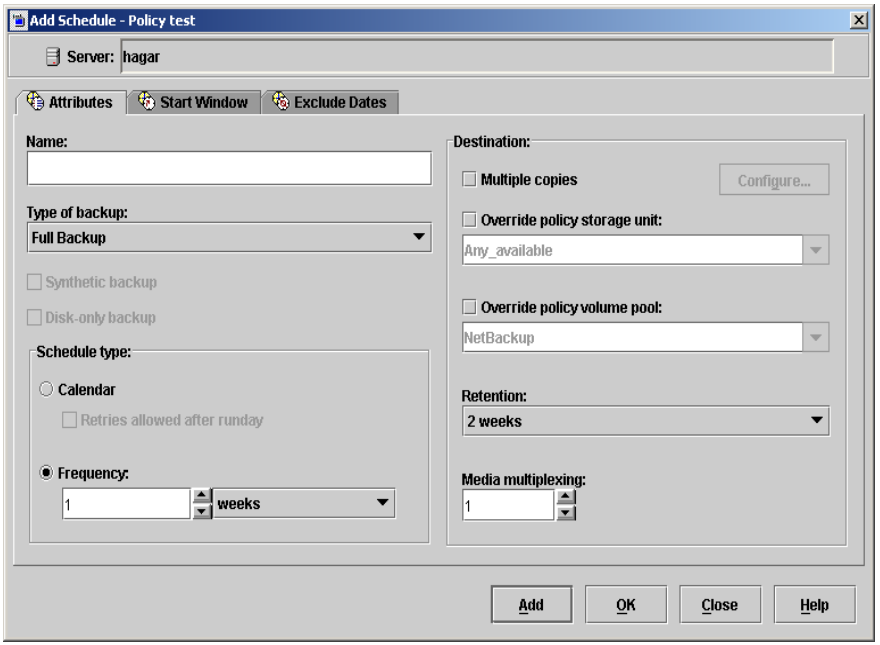
Example Settings for an Application Backup schedule

Type of Backup	Schedule settings	Description	Settings
Application Backup	Retention	The length of time backup images are stored.	2 weeks
	Backup Window	The time during which a NetBackup operation can be performed.	Sunday through Saturday 00:00:01 - 23:59:59

▼ To configure an automatic backup schedule

1. In the Policy dialog, click the **Schedules** tab.
2. Click **New**.

A dialog is displayed. The title bar shows the name of the policy to which you are adding the schedules.



3. Specify a unique name for the schedule.

4. Select the **Type of Backup**.

   For information on the types of backups available for this policy, see "Types of Backups."

   Refer to the following table for example settings for an Automatic Backup schedule.

Example Settings for a Automatic Backup Schedule

Type of Backup	Schedule settings	Description	Settings
Automatic Backup	Retention	The length of time to store the record of a backup.	2 weeks
	Frequency	Frequency determines how often a backup should be performed	every week
	Backup Window	The time during which a NetBackup operation can be performed.	Sunday, 18:00:00 - 22:00:00

Tip If you put multiple batch files in the same MS-SQL-Server policy, they will execute during each *Automatic Backup* session for that MS-SQL-Server policy. If you have a variety of SQL Server backup operations that you wish to have executed on different schedules, then consider creating multiple MS-SQL-Server policies with differing *Automatic Backup* schedules; and then assign each batch file to the policy that uses the desired *Automatic Backup* schedule.

Note For Automatic Backup schedules, the batch files must be installed on each machine in the client list.

5. Specify the other properties for the schedule as explained in "Schedule Properties."

6. If this is the last schedule, click **OK**.

Schedule Properties

Some of the schedule properties have a different meaning for database backups than for a regular file system backup. These properties are explained in the following table.

Other schedule properties vary according to your specific backup strategy and system configuration. Consult the *NetBackup System Administrator's Guide for Windows, Volume I* or *NetBackup System Administrator's Guide for UNIX, Volume I* for detailed explanations of the schedule properties.

Description of Schedule Properties

Property	Description
Type of backup	Specifies the type of backup that this schedule will control. The selection list shows only the backup types that apply to the policy you are configuring. For more information see the following section "Types of Backups."
Frequency	This setting is used only for scheduled backups, and not for user-directed backups. **Frequency** specifies the period of time that will elapse until the next backup operation can begin on this schedule. For example, if the frequency is seven days and a successful backup occurs on Wednesday, the next full backup will not occur until the following Wednesday.
Calendar	This setting is used only for scheduled backups, and not for user-directed backups. The **Calendar** option allows you to schedule backup operations based on specific dates, recurring week days, or recurring days of the month.

Description of Schedule Properties (continued)

Property	Description
Retention	**Frequency based scheduling** The retention period for an Application Backup Policy schedule refers to the length of time that NetBackup keeps backup images. For example, if your database is backed up once every Sunday morning, you should select a retention period of at least 2 weeks.The retention period for an Automatic Backup schedule controls how long NetBackup keeps records of when scheduled backups have occurred. The NetBackup scheduler compares the latest record to the frequency to determine whether a backup is due. This means that if you set the retention period to expire the record too early, the scheduled backup frequency will be unpredictable. However, if you set the retention period to be longer than necessary, the NetBackup catalog will accumulate unnecessary records. *Therefore, set a retention period that is longer than the frequency setting for the schedule.* For example, if the frequency setting is set to one week, set the retention period to be more than one week. **Calendar based scheduling** The retention period for an Application Backup Policy schedule refers to the length of time that NetBackup keeps backup images. The retention period for an Automatic Backup schedule controls how long NetBackup keeps records of when scheduled backups have occurred. However, this setting is not significant for calendar based scheduling.
Multiple copies	If you are licensed for the Inline Tape Copy feature and wish to specify multiple copies for your MS-SQL-Server policy, configure **Multiple copies** on the Application Backup schedule.

Types of Backups

Description of Backup Types for SQL Server

Application Backup	The Application Backup schedule enables user-controlled NetBackup operations performed on the client. At least one Application Backup schedule type must be configured for each MS-SQL-Server policy. The Default-Application-Backup schedule is automatically configured as an Application Backup schedule.
Automatic Backup	An Automatic Backup schedule specifies the dates and times when NetBackup will automatically start backups by running the batch files in the order that they appear in the file list. If there is more than one client in the MS-SQL-Server policy, the batch files are executed on each client.

Specifying the List of Backup Selections

The Backup Selections list in a database policy has a different meaning than for non-database policies. For example, in a Windows policy, the list contains files and folders to be backed up. In a database policy, the list contains batch files. NetBackup for SQL Server scripts are called batch files and are identified by the `.bch` extension.

Add batch files only if you are setting up a policy for automatic scheduling. All batch files listed in the Backup Selections list will be executed for the Automatic Backup schedules as specified under the **Schedules** tab. NetBackup will start backups by running the batch files in the order that they appear in the Backup Selections list.

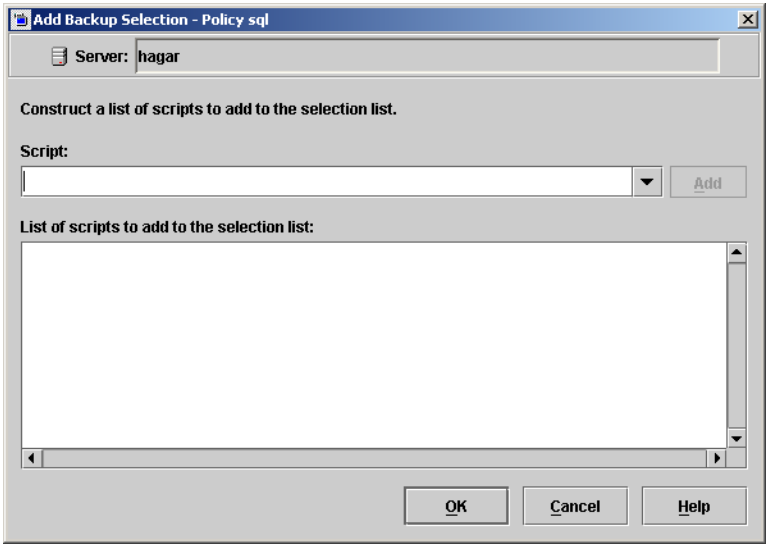
The following procedure starts from the Policy dialog. To see the Policy dialog, double-click the policy in the **Policies** list in the NetBackup Administration Console.

The optional Backup Selections list in an MS-SQL-Server policy lists batch files that describe the backup and restore operations you wish to start through manual or scheduled operations initiated from the NetBackup server. Batch files reside on the client and direct the operation of NetBackup for SQL Server and SQL Server. Refer to "Overview of Batch Files" on page 101 for details on how to create a batch file.

▼ To add batch files to the Backup Selections List

1. In the Policy dialog, click the **Backup Selections** tab.
2. Click **New**.

A dialog is displayed. The title bar shows the name of the policy to which you are adding the batch files.



3. Type the name of the batch file.

   Specify the full pathname for the batch file in the file list. For example, a path name for a batch file that will back up a database may be:

   install_path\NetBackup\DbExt\Mssql\bkup.bch

Note *install_path* refers to the directory where you installed the NetBackup software. By default, this directory is C:\Program Files\Veritas\.

   You can also specify just the file name if the batch file resides in the *install_path*\NetBackup\Dbext\Mssql\ directory.

4. Click **Add**.

5. To add more batch files, repeat step 3 and step 4.

6. When you are finished adding backup selections, click **OK**.

Adding Clients to a Policy

The client list is the list of hosts on which your NetBackup for SQL Server backups will be performed. A NetBackup client must be in at least one policy but can be in more than one.

This procedure starts from the Policy dialog. To see the Policy dialog, double-click the policy in the policy list in the left pane of the NetBackup Administration Console.

Requirements for clients

The following software must be installed on the client:

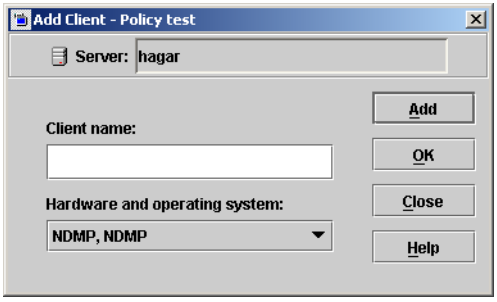
- SQL Server
- NetBackup client or server
- NetBackup for SQL Server
- the backup or restore script(s)

▼ To add clients to a policy

Note NetBackup for SQL Server must be installed on each client. Also, each batch file must be present on each client.

1. In the Policy dialog, click the **Clients** tab.
2. On the **Clients** tab, click **New**.

   The Add Client dialog is displayed. The title bar shows the name of the policy where you are adding clients.



3. In the **Client name** text box, type the name of the client that you are adding.

   If SQL Server is installed in a cluster, specify the virtual SQL Server name as the client name.

4. Choose the **Hardware and operating system** type and click **Add**.

5. If this is the last client, click **OK**. If you are going to add more clients, repeat step 3 and step 4.

Testing NetBackup for SQL Server Configuration Settings

After configuring the master server for NetBackup for SQL Server, test the configuration settings. For a description of status codes, refer to the *NetBackup Troubleshooting Guide for UNIX and Windows*.

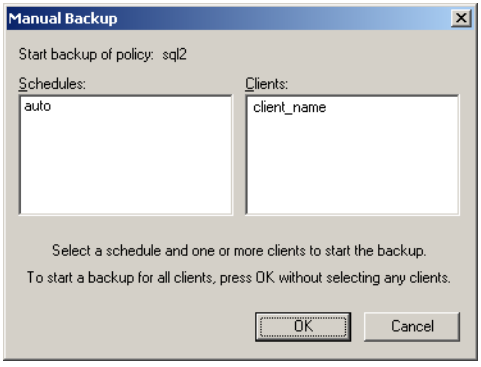
NetBackup Administration Console for Windows

Use this procedure to test a policy configuration from a Windows server or from the Remote Administration Console.

▼ To test the configuration settings on a Windows server

1. Log onto the server as administrator.
2. Start the NetBackup Administration Console.
3. In the left pane, click **Policies**. The policy list appears in the right pane.
4. Click on the policy you wish to test.
5. Choose **Actions** > **Manual Backup**.

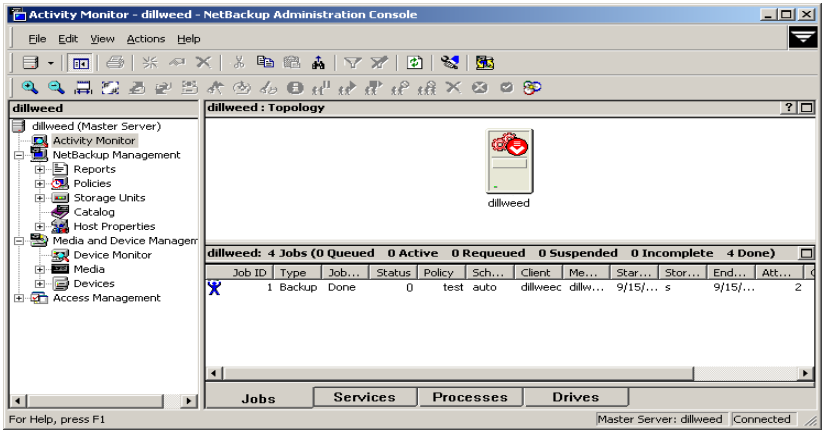
   The Manual Backup dialog is displayed.



   The Schedules pane contains the name of a schedule (or schedules) configured for the policy you are going to test. The Clients pane contains the name of the client(s) listed in the policy you are going to test.

6. Follow the instructions on the dialog.

7. Click **Activity Monitor** on the NetBackup Administration Console.



If the manual backup does not exit with a successful status, refer to the Troubleshooting chapter.

NetBackup Administration Console for UNIX

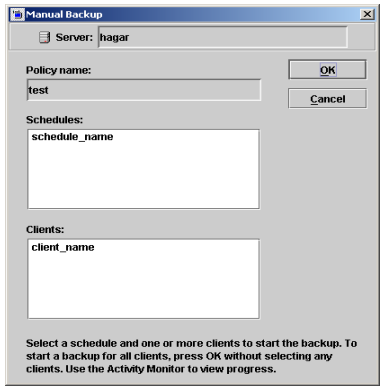
Use this procedure to test a policy configuration on the NetBackup Administration Console for UNIX.

▼ To test the configuration settings on a UNIX server

1. Log onto the server as root.
2. Start the NetBackup Administration Console.
3. In the left pane, click **Policies**.

   The right pane splits into an All Policies pane and a details pane.
4. In the All Policies pane, click the policy you wish to test.
5. Choose **Actions** > **Manual Backup**.

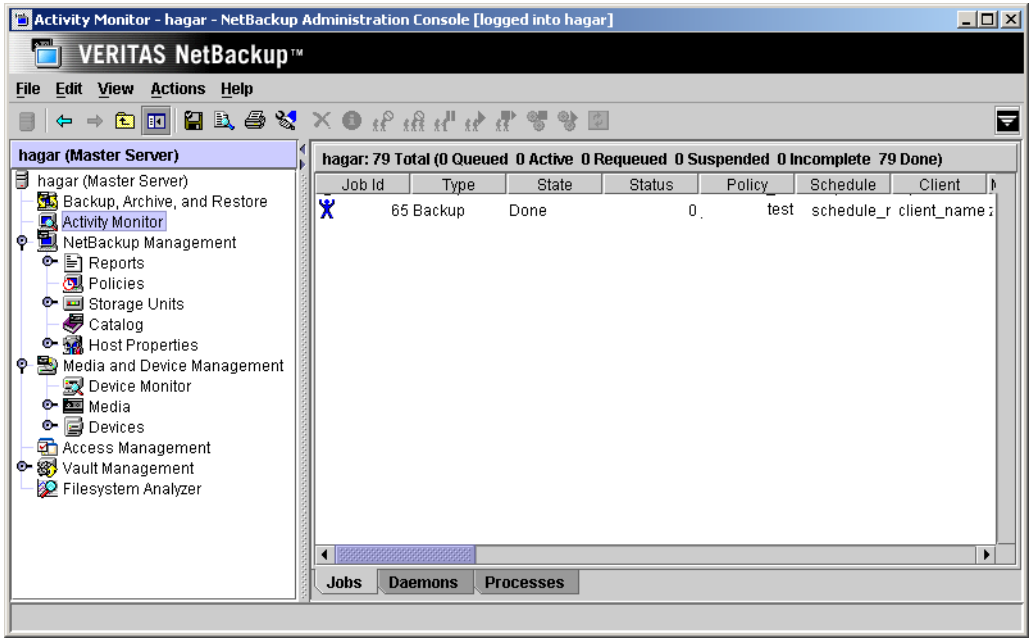
The Manual Backup dialog is displayed.



The Schedules pane contains the name of a schedule (or schedules) configured for the policy you are going to test. The Clients pane contains the name of the client(s) listed in the policy you are going to test.

6. Follow the instructions on the dialog.

7. Click **Activity Monitor** on the NetBackup Administration Console.



If the manual backup does not exit with a successful status, refer to the Troubleshooting chapter.

Configuring for Multi-Stream Operations

SQL Server supports backing up and restoring databases through multiple data streams, which are called stripes. This feature is also supported by NetBackup for SQL Server. You can archive the streams to separate devices or you can multiplex them to the same device. Multiplexing means that you use just one storage device as the target for all of the data streams.

The following sections describe how to configure for non-multiplexed and multiplexed backups. In addition, see the section "Starting Multi-streamed Operations" on page 124 for information on how to start striped backups and restores after you have configured NetBackup.

Configuring for Non-Multiplexed Backup Stripes

When backup striping is used, all streams must be available before the backup or restore can proceed. Also, the number of stripes specified in the restore must match the number specified during the backup. Therefore, if you wish to use backup striping without multiplexing you must:

- Disable multiplexing on the *Application Backup Policy* schedules that you will use for striped backups or restores. This is necessary because all the streams must be simultaneously available before a restore can proceed. You can do this by setting the Media Multiplexing property in your *Application Backup Policy* schedule to 1.
- Ensure that the storage unit has as many drives as you will have stripes.

In addition, you must ensure that backups and restores are scheduled so that enough drives are available at the time you will be performing striped backups or restores.

Configuring for Multiplexed Backup Stripes

If you use backup stripes and you want the data streams to be stored to a single device, you must do the following.

- Set the Media Multiplexing property in your *Application Backup Policy* schedule to the number of backup stripes that you plan to use.
- Set the Maximum Multiplexing per Drive property for the storage units which you associate with your *Application Backup Policy* schedule to the number of stripes that you plan to use.

SQL Server Configuration

This section contains information on configuring the SQL Server for NetBackup operations.

DBMS (Database Management System) Privileges

When you install SQL Server, you can elect to use either integrated or standard security. Integrated security refers to the usage of Windows authentication in lieu of standard SQL Server-based logins.

Note Microsoft recommends that you use integrated security because Windows logins, unlike SQL Server-based logins, can be traced using standard Windows security tools. NetBackup for SQL Server supports both integrated and standard security for any level of SQL Server.

If you choose integrated security, then SQL Server ignores the userid and password that you provide in the Set database login parameters dialog of the NetBackup Database Extension GUI. Instead, security validation to perform backup and restore operations and for querying the master database will be based upon the SQL Server privileges that have been granted to the Windows account. The batch file userid and password keywords are also ignored. When integrated security is used SQL Server operations generated by NetBackup for SQL Server will always be validated against the Windows userid.

If you choose standard security, then you must supply NetBackup for SQL Server with a SQL Server-based userid and password. This userid and encrypted password are stored in the registry under `HKEY_CURRENT_USER\SOFTWARE\VERITAS\NETBACKUP\NetBackup for SQL Server\`. In order to perform an operation with standard SQL Server security, NetBackup for SQL Server looks up the SQL Server userid and password from the registry.

Note Since NetBackup for SQL Server does backup and restore commands and issues select statements against the master database, you should review the SQL Server user documentation in order to determine exactly what privileges are required by the user accounts that you establish for NetBackup for SQL Server.

Setting the SQL Server Login for Scheduled Operations

When you use the NetBackup Database Extension GUI, SQL Server security is satisfied based upon either your current Windows account, if you are using integrated security, or a SQL Server-based userid and password. For information on integrated and standard security, see the previous section “DBMS (Database Management System) Privileges”.

When the NetBackup scheduler is used, backup and restore operations are launched from a daemon process called the NetBackup Client Service. You must ensure that the Client Service has SQL Server privileges to perform backup and restore operations. If you are using integrated security, then you must ensure that the NetBackup Client Service specifies a Windows account that has been granted SQL Server administrative privileges.

Note If your site does not place any SQL Server security restrictions on the right to back up databases then the steps described in the next two sections will be unnecessary.

Authorizing Scheduled Operations (for Sites with SQL Server Security Restrictions)

If you will be using NetBackup to perform automatic SQL Server backups and if your site places SQL Server security restrictions on the right to back up databases, then an account must be set up which the NetBackup Client Service can use to login to SQL Server. The account must have sufficient SQL Server privileges to perform a backups.

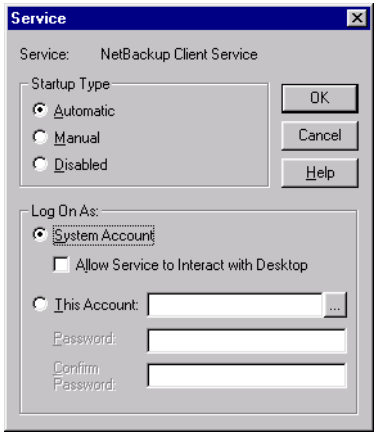
▼ To set up the NetBackup Client Service to login to SQL Server

1. If you are using standard SQL Server security, first map a Windows account to a standard SQL Server userid. (See "How to Use Standard SQL Server Security" on page 49.)

2. Open **Services** applet from the Windows Control Panel.

3. Double-click on **NetBackup Client Service** from the Service panel to bring up the Service dialog box for the NetBackup Client Service.

4. In the **Log On As** group, select **This Account**.



5. Click the browse button (...).

   The **Add User** dialog box is displayed.

6. Select a user account.

7. Click **Add**.

8. Type the password and the password confirmation.

9. Click **OK**.

Note If you later change the password associated with the account name entered in step 8, then you must also indicate the new password for the NetBackup Client Service, then restart the NetBackup Client Service. If you do not do this, NetBackup scheduled backup operations will fail.

---

10. Log on to Windows with the Windows account selected in step 6. Then perform the procedure described in "Setting the Database Login Parameters" on page 63. The database login parameters that you choose will be selected for SQL Server backups launched from the NetBackup Scheduler.

11. Stop and restart the NetBackup Client Service from the service dialog box.

How to Use Standard SQL Server Security

If you are using standard SQL Server security, then you must ensure that the NetBackup Client Service is started with a Windows userid that is associated with a standard SQL Server userid that has sufficient administrative privileges to perform backups and restores.

▼ To associate a Windows userid with a SQL Server userid

1. Log on to your Windows host using the Windows account which you intend to use for the NetBackup Client Service.

2. Start the NetBackup Database Extension GUI and go to the DBMS login parameters window. Type the standard SQL Server login name and password into the edit boxes provided. The values that you supply will be encrypted in a registry entry that is associated with the current Windows account.

3. Follow the instructions in previous section "Authorizing Scheduled Operations (for Sites with SQL Server Security Restrictions)" to set the login account for the NetBackup Client Service.

Protecting Files and Filegroups

If you plan to back up database files or filegroups, then you will be responsible for creating the files and filegroups for your databases and for placing individual database components into them. NetBackup places a restriction on the layout of your database in order to successfully perform backup and restores of database files and filegroups.

Before attempting a file and filegroup backup, you must ensure no table is placed into a filegroup which is different than any one of its indices.

For example, the layout as indicated by the following Transact SQL statements should not be used:

```
use master
CREATE DATABASE MultiFileDB
ON
PRIMARY ( NAME = FileX,
          FILENAME = 'd:\mssql\data\FileX.mdf'),
FILEGROUP AltGroup
( NAME = AltGroupFil,
  FILENAME = 'd:\mssql\data\AltGroupFil.ndf')
GO
use MultiFileDB
CREATE TABLE Table1 (col1 char(10),col2 char(10), col3 char(10)) on AltGroup
go
create unique clustered index index4 on Table1 (col2)
go
```

Notice in this example, Table1 has been placed in filegroup AltGroup but its index is placed (by default) in the primary filegroup.

If you do place a table into a different filegroup than one of its indices and use NetBackup for SQL Server to back it up, you may fail with the following SQL Server error message:

```
Database file <file name> is subject to logical recovery and must be among
the files to be backed up as part of the file or filegroup backup.
```

Recovery Considerations for Files and Filegroups

To ensure that you will be able to successfully restore a database from file and filegroup backups, it is important to ensure that you always have backups of a full set of files and filegroups that constitute the entire database as well as transaction log backups that span the entire period of time over which the backups were taken. To ensure that you maintain the unbroken sequence of transaction log backups, it is essential to perform a transaction log backup following every file or filegroup backup. (If you backup several files or filegroups at once, then you only need to backup the transaction log after the last such backup.) If the transaction log is not backed up, then SQL Server will not allow you to use the file or filegroup backup for a restore operation until the transaction log has been backed up and is thus available in the recovery sequence.

It is also important to understand that SQL Server does not record the creation of files or filegroups in the transaction log. Therefore, after you add either a file or a filegroup to the database, you must immediately back them up. In addition, to ensure that NetBackup for SQL Server will select the correct recovery set following the backup of a newly created filegroup, you are also advised to back up all filegroups in the database. For the same consideration, following the creation a database file, you should back up all of the files in the filegroup to which it belongs.

Recovery Factors

This section is provided to assist you in defining a recovery plan which is suitable to your application environment. Much of this information is based on Microsoft's *SQL Server Books Online*. Refer to that resource for a more inclusive discussion.

Transaction Logs

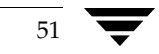
SQL Server maintains a write-ahead transaction log for each database. This log helps to maintain database updates in cache memory to ensure that data is not written to disk before it has been committed. Database writes occur as a part of the checkpoint procedure. Checkpoint frequency is determined by SQL Server based upon the "recovery interval" which is a configuration parameter indicating the maximum time interval that can be tolerated during a system restart. When checkpoint occurs the portion of the transaction log that is no longer needed for system restart becomes inactive and is optionally truncated, depending upon the recovery strategy in place, as described in the next section.

If the transaction log is not truncated by the checkpoint procedure, then it can be backed up and used for point-in-time recovery, failure from disk crash, or move and copy operations.

Recovery Strategies

SQL Server 2000 provides three basic levels for database recovery which have different implications for both backup performance and for the granularity of recovery. These levels are:

- *Simple*. With this method the inactive portion of the transaction log is not retained beyond the database checkpoint, this method provides for minimal usage of log space. However, the database can only be restored to the last full backup. Transaction log restores, including point in time recovery and named transaction recovery are not supported. In addition, maximum performance is provided for bulk operations, such as (Create Index, Select Into, and Bulk Copy) because they are not logged.



- *Full*. Using this method, the inactive portion of the transaction log is retained until it is truncated, which normally occurs when it is backed up. The transaction log can then be used to stage a recovery either to a point in time or to a named transaction. The Full Recovery model provides maximum recoverability but it uses the most log space and does not provide maximum performance for bulk operations.
- *Bulk-Logged*. This method is identical to the Full Recovery model except that bulk operations are not logged and thus cannot be recovered.

SQL Server 7.0 provides similar recovery as SQL Server 2000 except that functionality is defined by two attributes that are provided via the database options menu. These are:

- *Truncate log on checkpoint*. Enables "simple" logging, i.e., the inactive portion of the transaction log is truncated on checkpoint.
- *Select Into/Bulk Copy.* Bulk operations are not logged.

Backing up the Transaction Log

Transaction logs can be backed up in SQL Server 2000 for databases which are set in either full or bulk-logged mode. In SQL Server 7, transaction logs can be backed up unless "truncate log on checkpoint" has been checked. By default a transaction log is truncated after it has been backed up. However, it is not truncated following a full database or differential backup.

The main factors in deciding how frequently to backup a transaction log would be

- Conservation of log space.
- How close to the failure point you must be able to recover in case of a disk crash.

During peak periods in a high transaction environment, it may not be unusual to back up the transaction log on an hourly basis.

Database Differential Backup

Unlike the transaction log backup, the database differential backup is a backup of the database. The differential includes all of the changes made to the database since the last full backup. If you have made several differential backups since the last full backup, then to accomplish a full restore to the end of the last differential backup, you would only need to restore the last full database, followed by the last differential. You would not need to restore any of the intermediate differentials.

Although database differential backups are supported with both SQL Server 7.0 and 2000, the performance of SQL Server 2000 differential backups are especially high in performance because SQL Server maintains the list of changed data pages in a bit map, thereby avoiding the need to read unchanged pages.

A typical backup procedure may use full database, differential, and transaction log backups in ascending order of frequency. For example the full database backup may be taken bi-weekly, the differential may be taken nightly, and the transaction log backup may be made as frequently as every hour for either mission critical or high volume applications.

File and Filegroup Backups

In addition to full, database differential, and transaction log backups of a database, SQL Server also supports backing up database filegroups and files as well as filegroup differentials. Since a filegroup is composed of one or more database files, backing up the constituent files of a filegroup is logically equivalent to backing up the filegroup itself.

Filegroup and file backups would commonly be used in a tightly architected application in which physical disk locations were mapped to logical objects, such as tables and indexes. There are two types of factors which may lead to the choice to use file and filegroup backups in this type of environment.

- Some portions of the database, which may be distinguished by such factors as volatility or mission criticality, may be recognized as having to be backed up more frequently than other portions.
- The database may be so large that the time required for a full database backup could not fit in the allocated time window. Thus it may be more viable to do a full backup of one or more files or filegroups on a rotating basis.

The filegroup and file backup also carries the advantage on restore that, in case of disk failure, it would be possible to recover just the failed unit without restoring the entire database.

In order to use filegroup and file backups you must maintain backups of the transaction log. For example, to perform a full database restore using filegroups and files, you would be required to restore all of the constituent filegroups and files in addition to all of the transaction log segments starting from the point at which the first component backup was taken up until a point in time following the last component backup.

Database Recovery

During the restore process, a database goes into “loading mode” until the restore command is executed against the database with the “recovery” option. Prior to placing the database into recovery mode all of the restore commands would be executed using the “WITH NORECOVERY” clause. By specifying NORECOVERY, it is possible to continue to stage additional restore statements to bring the database up to the desired state. The database becomes usable again after the last restore statement has been applied with recovery.

Staging Recovery

NetBackup for SQL Server keeps track of the backups you have performed and when you performed them. You can display the backup history by opening the "Restore Microsoft SQL Server Objects" dialog (see "Restore Microsoft SQL Server Objects" on page 224)Restore Microsoft SQL Server Objects. This window depicts all of the SQL Server backup images within the parameters that you specify. The images are displayed in tree-form based on SQL Server logical groupings for databases, database differentials, filegroups, filegroup differentials, files, and transaction logs.

When you select a transaction log image in the restore tree, NetBackup for SQL Server automatically searches for a set of images which are sufficient for staging a full database recovery. The recovery set consists of the selected transaction log image plus additional images which could reconstitute the database to a recovered state. The full set can be comprised based on any one of the following sets:

1. Full database backup set: A full database image, an optional database differential image, and one or more transaction logs.

2. File and filegroup set: A set of filegroup and/or file images which constitute the database, an optional filegroup differential image for any of the filegroup images, and one or more transaction logs.

3. Database and filegroup differential set: A full database image, a filegroup differential image for each filegroup in the database, and one or more transaction logs.

If a full recovery set is found, then all of the composite images will be checked automatically and a pop-up message will appear stating the existence of the restore set. You can then initiate a staged recovery (or generate a batch file for staged recovery) by clicking on the **OK** button with the option "stage a full recovery" selected.

Backing Up SQL Server in an Environment with Log Shipping

Log shipping is a SQL Server feature that may be employed to enhance the overall availability of your installation. It uses a primary server, which contains the active database, a monitor, and one or more secondary servers. Under log shipping, copies of the transaction log are supplied to the secondary servers on an on-going basis to the secondary servers. This allows each secondary server to be in a standby state in case the primary goes offline.

Many sites also use the secondary server to off-load certain activities from the primary in order to minimize its load. *However, a backup must not be performed on a secondary (or standby) server.* Databases must always be backed up on the primary server and restored on the primary server. This requirement is based on Microsoft SQL Server restriction that

is outlined in Microsoft Knowledge Base Article 311115. If an attempt is made to perform a backup on the secondary server, the result cannot be predicted. However, you may see a message in the dbclient log similar to the one below:

```
16:33:26 [1208,2348] <16> CODBCaccess::LogODBCerr: DBMS MSG - ODBC
message. ODBC return code <-1>, SQL State <37000>, Message Text
<[Microsoft][ODBC SQL Server Driver][SQL Server]Database 'Mumbo'
is  in warm-standby state (set by executing RESTORE WITH STANDBY)
and cannot be backed up until the entire load sequence is
completed.>
```

Performance Factors

Backup performance can be influenced by your hardware environment as well as settings in SQL Server, the NetBackup master server, and NetBackup for SQL Server. This section discusses the most prominent factors in SQL Server and NetBackup for SQL Server. The availability of buffer space has a definite influence on how fast backups run.

Buffer Space Parameters

There are three parameters you can use to increase buffer space in SQL Server.

MAXTRANSFERSIZE

The MAXTRANSFERSIZE parameter can be set for each backup or restore operation. It ranges in size from 64 kilobytes to 4 M bytes. The default is 64 kilobytes. MAXTRANSFERSIZE is the buffer size used by SQL Server for reading and writing backup images. Generally, you can get better SQL Server performance by using a larger value.

Set the MAXTRANSFERSIZE for your NetBackup for SQL Server session using the Client Options dialog. Otherwise, you can use the MAXTRANSFERSIZE parameter to set this value when you create a batch file manually. For more details on the MAXTRANSFERSIZE parameter, see “Overview of Batch Files” on page 101.

BLOCKSIZE

The BLOCKSIZE parameter can be set for each backup operation. For restores, NetBackup for SQL Server automatically chooses the same size that you backed up with. BLOCKSIZE ranges from 512 bytes to 64 kilobytes. The default is 512 bytes. BLOCKSIZE is the incremental size that SQL Server uses for reading and writing backup images.

Set the BLOCKSIZE for your NetBackup for SQL Server session using the Client Options dialog. Otherwise, you can use the BLOCKSIZE parameter to set this value when you create a batch file manually. For more details on the BLOCKSIZE parameter, see "Overview of Batch Files" on page 101.

NUMBER OF BUFFERS PER STRIPE

The NUMBER OF BUFFERS PER STRIPE parameter affects buffer space availability. This parameter is set in the NetBackup for SQL Server administration GUI for each operation. NetBackup for SQL Server uses this parameter to decide how many buffers to allocate for reading or writing each data stream during a backup or restore operation. By allocating a greater number of buffers, you can affect how quickly NetBackup for SQL Server can send data to the NetBackup master server.

The default value for the NUMBER OF BUFFERS PER STRIPE is 1. For operations that you start from the NetBackup Database Extension GUI, you can change this value through the Client Options dialog. See "Client Options" on page 220. Otherwise, you can use the BUFFERS parameter to set this value when you create a batch file manually. For more details on the BUFFERS parameter, see "Overview of Batch Files" on page 101.

Backup Stripes

You can divide your backup operation into multiple concurrent streams in any situation in which SQL Server dumps data faster than your tape drive is capable of writing. Although NetBackup supports multiplexing many stripes to the same drive, you generally will realize performance gains only if you use the same number of tape drives as streams.

To divide a backup operation into multiple stripes, use the **How Many MS SQL Server Stripes?** edit box, which appears in the Backup Microsoft SQL Server Objects dialog of the NetBackup Database Extension GUI.

Shared Memory Usage

If NetBackup Server is installed on the same host as NetBackup for SQL Server, then you will obtain optimal performance by using shared memory for data transfer instead of sockets. Shared memory is the default for this configuration and will be used unless you create a `install_path\NetBackup\NOSHM` file.

Alternate Buffer Method

NetBackup for SQL Server supports an alternate buffer method which optimizes CPU usage by allowing NetBackup and SQL Server to share the same memory buffers without transferring data between them. This method is chosen automatically for backups if all of the following conditions apply:

- NetBackup shared memory is in use.
- The backup is non-multiplexed.
- The NetBackup buffer size equals the SQL Server blocksize.

  The default NetBackup buffer size is 64 kilobytes, but this may be overridden by the value contained in *install_path*\db\config\SIZE_DATA_BUFFERS. To start your backups with the desired SQL Server blocksize, you can either set this parameter in the Client Options dialog of the NetBackup Database Extension GUI or you can adjust it directly using the BLOCKSIZE option in your batch file.
- NetBackup for SQL Server must be started with the same account as the NetBackup Client service.

  Backups started with the NetBackup scheduler are started with the NetBackup Client service so the same account is already in use. However, if you start an SQL Server backup through NetBackup for SQL Server or through dbbackex, then your logon account must be the same as the NetBackup Client service account in order for your backups to be candidates for the alternate buffer method.
- Alternate buffer method is only available for tape-based backups.

In addition to the above conditions, the alternate buffer method for restore also requires that the following conditions be met:

- The version of SQL Server being restored must be SQL Server 2000.
- The backup must have been made with alternate buffer method.

You can verify that the alternate buffer method is being used by looking for the words `Using alternate buffer method`, which appear in the dbclient log and the progress report.

Using NetBackup for SQL Server 4

After completing the installation and configuration, you can use the NetBackup Administration Console to start SQL Server backups or use the NetBackup for SQL Server interface to initiate backups and restores.

This chapter describes how to perform user-directed operations for SQL Server on a Windows client. You can initiate NetBackup operations using the command line interface program, `dbbackex`, or the NetBackup for SQL Server interface.

Using dbbackex

dbbackex can be executed from a Windows MS-DOS command prompt as

```
install_path\NetBackup\bin\dbbackex -f file [-p policy][-u userid][-pw password]
[-s server][-np]
```

where:

file is the name of the batch file, (see "Running NetBackup Batch Files" on page 120) which describes the operations you wish to start.

policy is the MS-SQL-Server type policy, used by NetBackup for the operations specified in the batch file.

userid is the SQL Server userid for logging into the database management system.

password is the SQL Server password for logging into the database management system.

server is the name of the host for the NetBackup master server that you wish to backup to or restore from.

-np tells dbbackex not to create a message box to indicate the operation status when it has completed.

Note Any of the options can be delimited with double quotation marks. Use delimiters, for example, if the file name contains spaces.

The *policy* parameter is ignored for restore operations as the NetBackup server can retrieve the dump file based entirely on the image names specified in the batch file for each restore. Policy is used for backing up databases. If *policy* is omitted, then NetBackup server uses the first active SQL Server policy that it finds in its policy list for carrying out all of the backup operations specified in the batch file.

If *server* is omitted, then the client will use the default server according to the Windows NetBackup client configuration. See the *NetBackup Backup, Archive, and Restore Getting Started Guide* for more information.

Tip If you are concerned about protecting login passwords for SQL Server, do not use the -u or -pw parameters. By omitting these parameters, you can force that NetBackup for SQL Server to read the default SQL Server login data from an encrypted file as described in "Starting the NetBackup Database Client Graphical User Interface for the First Time" on page 62.

Unless you specify -np, a message is displayed when dbbackex completes. This message will tell you how many operations in the batch file were successful and how many failed.

Using Client-based Schedulers with dbbackex

`dbbackex` allows you to employ your choice of client-based schedulers to automatically initiate NetBackup for SQL Server operations. Two examples follow.

- The Windows Scheduler. This scheduler is initiated through the MS-DOS `at` command. Instructions for using the Windows Scheduler are provided in the Microsoft Windows online documentation.
- The SQL Server Scheduler. This scheduler is closely integrated with SQL Server. It is accessed through the Microsoft SQL Server Enterprise Manager, from the server menu. Online help is provided.

One distinct advantage of the SQL Server Scheduler is that you can create scripts for database maintenance operations which will be initiated as a result of database events that you define. For example, you can create a script that initiates `dbbackex`, telling it to back up a particular transaction log, and then create an alert which invokes that script when the transaction log for this database becomes full.

Tip If you use `dbbackex` through a client-based scheduler, specify the `-np` option to ensure that a message box is not generated. However, before using the scheduler with `dbbackex`, try the `dbbackex` syntax on the console *without the* `-np` *option* in order to test for the successful completion of the batch file that you have created for your operation.

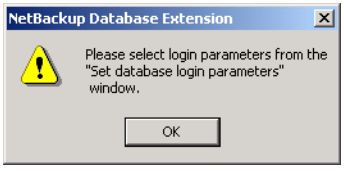
Using the NetBackup Database Client Graphical User Interface

These instructions describe how to use the NetBackup Database Extension GUI to perform NetBackup operations on SQL Server.

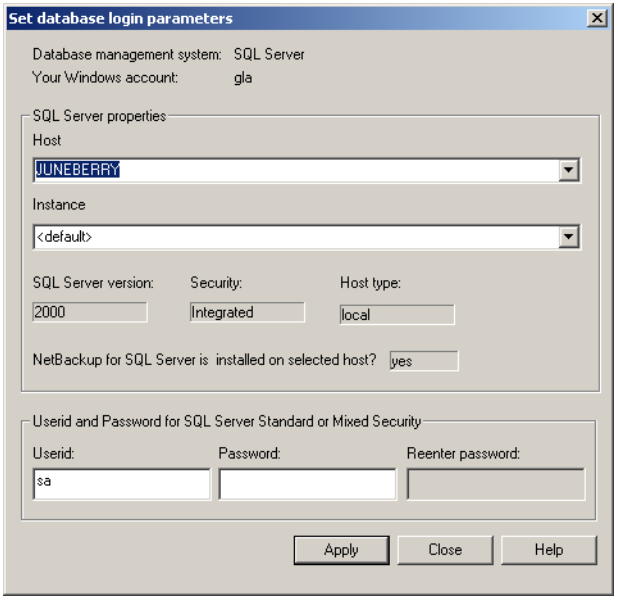
Starting the NetBackup Database Client Graphical User Interface for the First Time

1. Choose **Start** > **Programs** > **VERITAS NetBackup** > **NetBackup MS SQL Client**.

   The first time that you open the NetBackup Database Extension GUI on a new user account, the following message box is displayed.



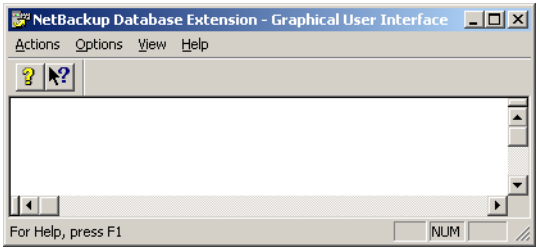
   After clicking **OK**, the Set database login parameters dialog is displayed.



2. Select the SQL Server host and instance that you want to log into.

3. Click **Apply** and **Close**.

   The NetBackup Database Extension GUI is displayed.



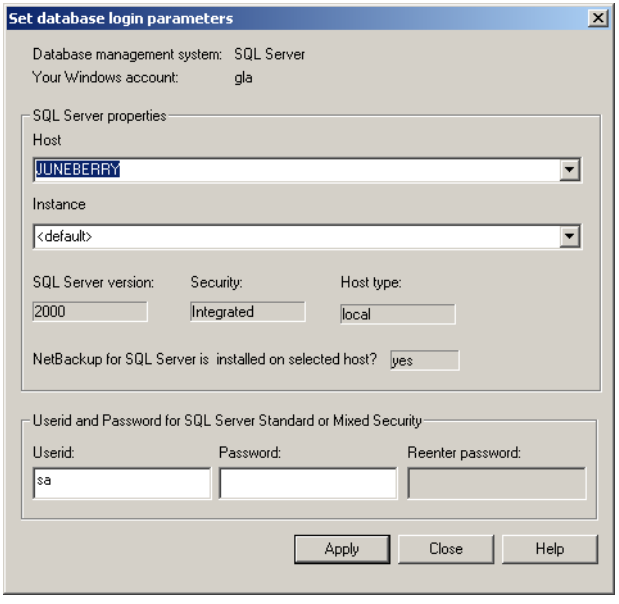
Note For subsequent sessions, the NetBackup for SQL Server agent will remember the login parameters you provided for that Windows account.

Setting the Database Login Parameters

Use this procedure to set which SQL Server host and instance that you want the NetBackup MS SQL Client to access when it logs into SQL Server.

1. Choose **Options** > **Set DBMS login parameters**.

   The Set database login parameters dialog is displayed.



2. Use the **Host** drop-down list to select the SQL Server host and the **Instance** drop-down list to select the SQL Server instance. These drop-down lists can also be manually edited if the host and instance that you want do not appear in the drop-down lists. You can designate the default instance either by setting the **Instance** box to <default> or to empty (not spaces).

3. If the SQL Server host and instance use standard or mixed security, then you also need to set the userid and password in the **Userid**, **Password**, and **Reenter password** boxes.

4. Click **Apply** to save your changes and **Close** to exit from the dialog.

NetBackup Backup Operations

Backing up a Database

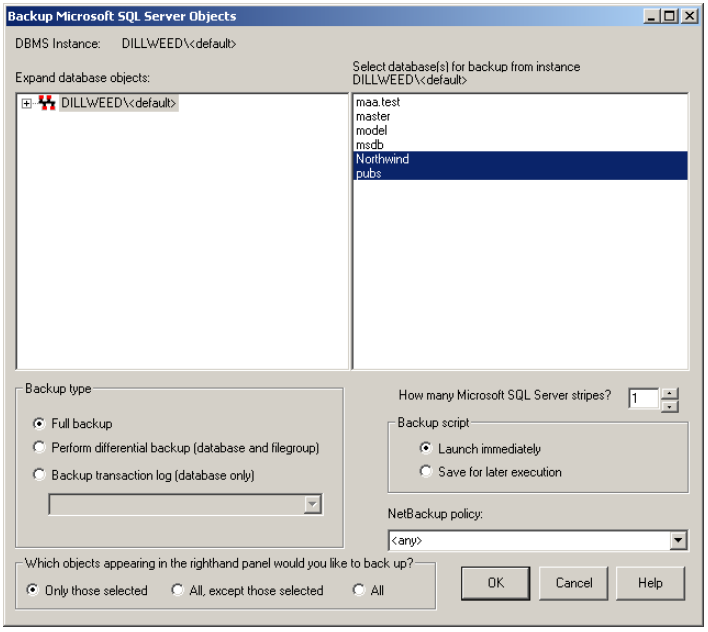
▼ To back up a database

1. To change the host and instance you wish to access, refer to "Setting the Database Login Parameters" on page 63.

2. Choose **Actions** > **Backup**.

   The Backup Microsoft SQL Server Objects dialog is displayed.

3. In the right pane, select one or more databases that you want to back up. Shift + click or Ctrl + click to select more than one database.



4. If you would like to backup the database with the differential option, then select **Perform differential backup**.

5. Click **OK**.

6. When prompted to start the backup, click **Yes**.

Backing up a Transaction Log

▼ To back up a transaction log

1. Before starting a transaction log backup, the database administrator should perform the following actions on the SQL Server.
   - If SQL Server 2000 is installed set the **Recovery Model** setting to either **Full** or **Bulk-logged**.
   - If SQL Server 7.0 is installed, set the **Truncate Log On Checkpoint database** option to off. This is an option on the SQL Server interface that applies to the database.

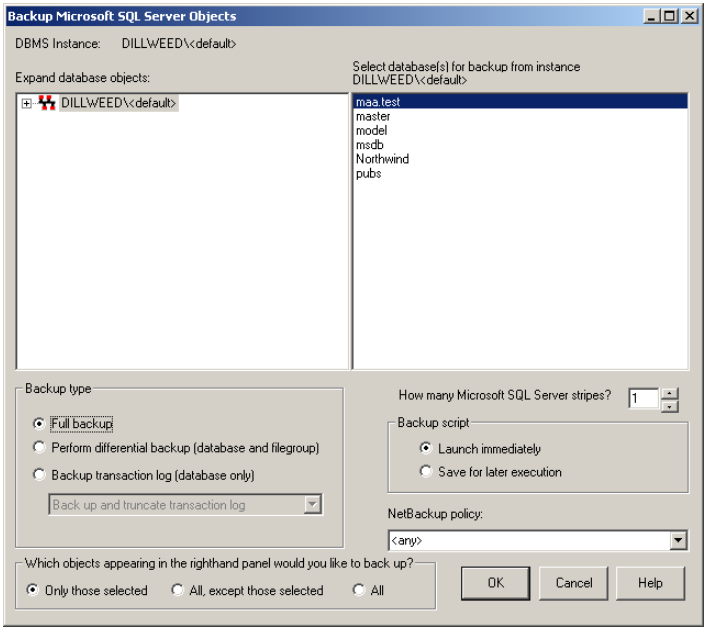
Caution You are responsible for ensuring that the entire sequence of transaction logs generated following any database backup are maintained on the same NetBackup server. This means that they should all be backed up to the same facility and that none should be allowed to expire before the others. NetBackup for SQL Server requires that you follow these guidelines in devising your backup strategy in order to provide maximum assistance in restoring your databases.

2. To change the host and instance you wish to access, refer to "Setting the Database Login Parameters" on page 63.

3. Choose **Actions** > **Backup**.

   The Backup Microsoft SQL Server Objects dialog is displayed.

4. In the right pane, select one or more databases whose transaction logs you want to back up. Shift + click or Ctrl + click to select more than one database.

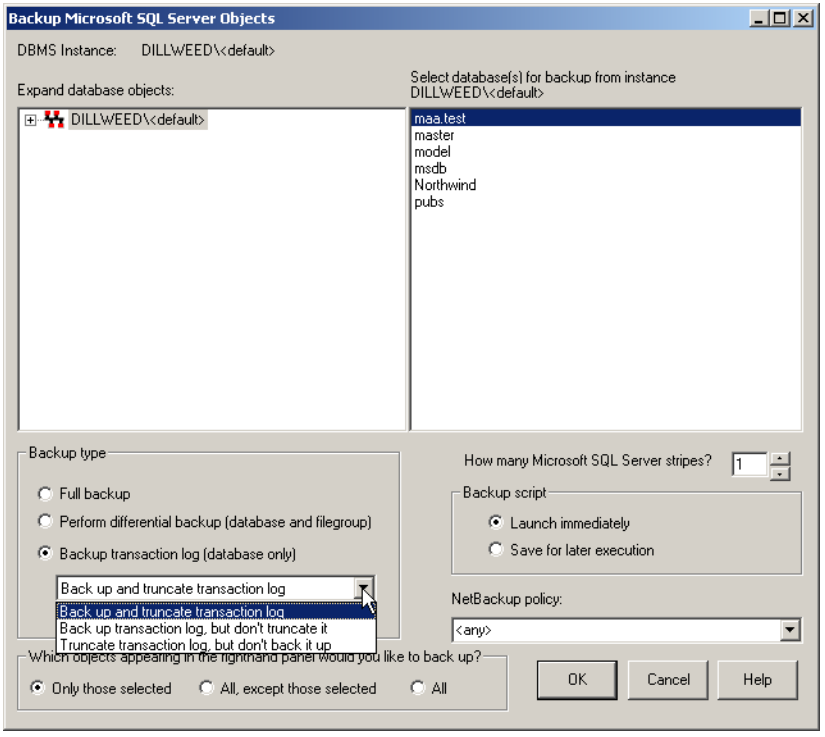


5. Select **Backup transaction log**.

6. From the drop-down list, select the desired transaction log option. For more information, refer to the following table.

Transaction log backup options

Back up and truncate transaction log	Back up the transaction log and remove the inactive part of the transaction log.
Back up transaction log but don't truncate it	Back up a transaction log without truncating it.
Truncate transaction log but don't back it up	Remove the inactive portion of the transaction log without backing it up.



7. When prompted to start the backup, click **Yes**.

Backing Up a Database Filegroup

Caution Do not use NetBackup for SQL Server to perform filegroup backups for a database that contains any table in which any one of the indices is in a different file or filegroup than its data. See "Protecting Files and Filegroups" on page 50.

▼ To back up a database filegroup

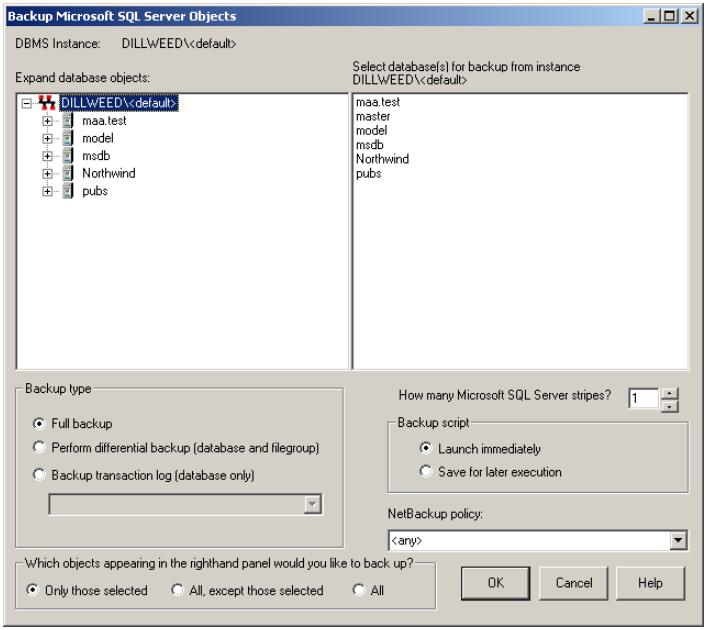
1. To change the host and instance you wish to access, refer to "Setting the Database Login Parameters" on page 63.

2. Choose **Actions > Backup**.

   The Backup Microsoft SQL Server Objects dialog is displayed.

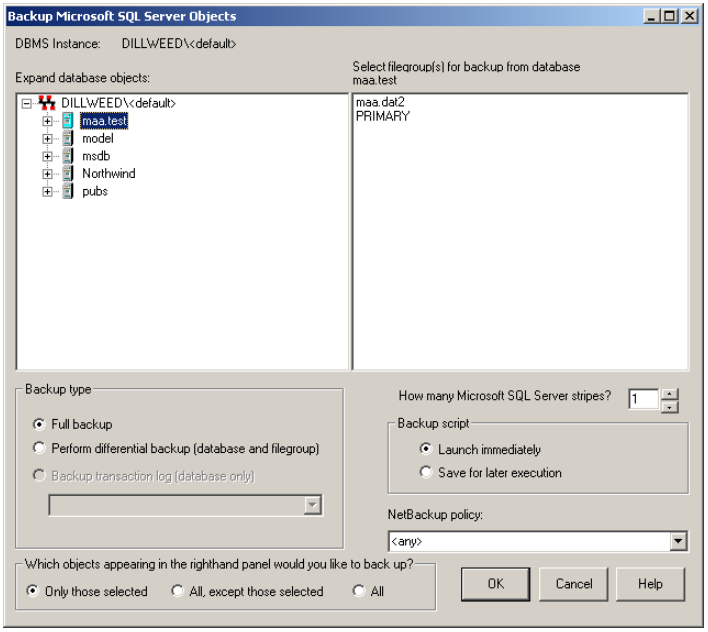
3. In the left pane, expand the instance name.

   The database names are displayed below the instance name.



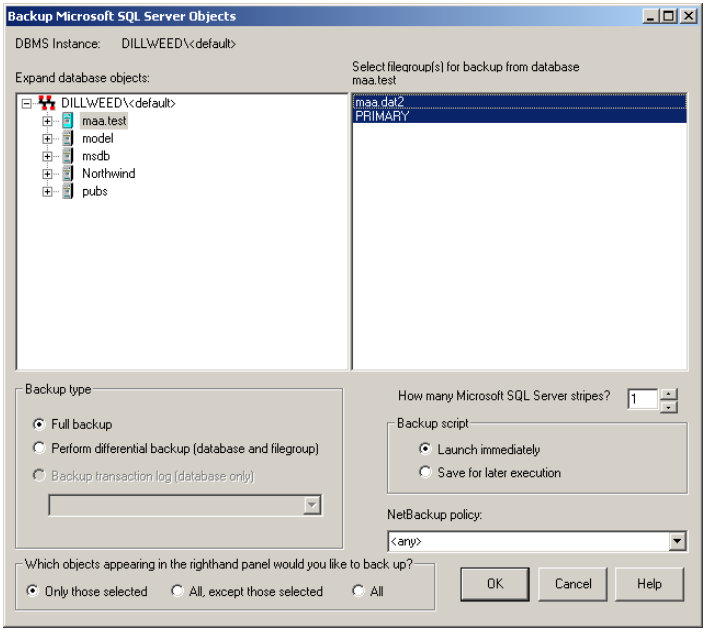
4. In the left pane, select a database whose filegroups you would like to back up.

In the right pane, the filegroups belonging to the selected database are displayed.



5. In the right pane, select one or more filegroups that you want to back up.

Shift + click or Ctrl + click to select more than one filegroup.



6. Click **OK**.

7. When prompted to start the backup, click **Yes**.

Backing Up a Database File

Caution Do not use NetBackup for SQL Server to perform file backups for a database that contains any table in which any one of the indices is in a different file or filegroup than its data. See "Protecting Files and Filegroups" on page 50.

▼ To back up a database file

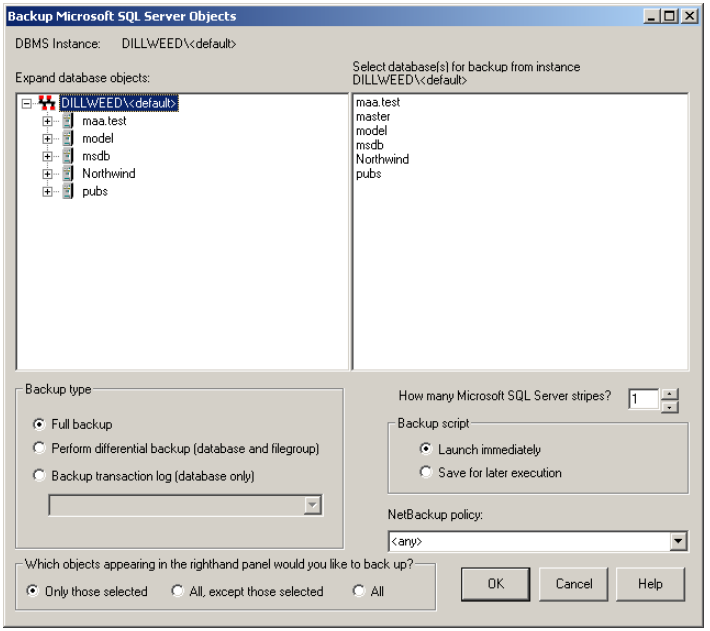
1. To change the host and instance you wish to access, refer to "Setting the Database Login Parameters" on page 63.

2. Choose **Actions > Backup**.

   The Backup Microsoft SQL Server Objects dialog is displayed.

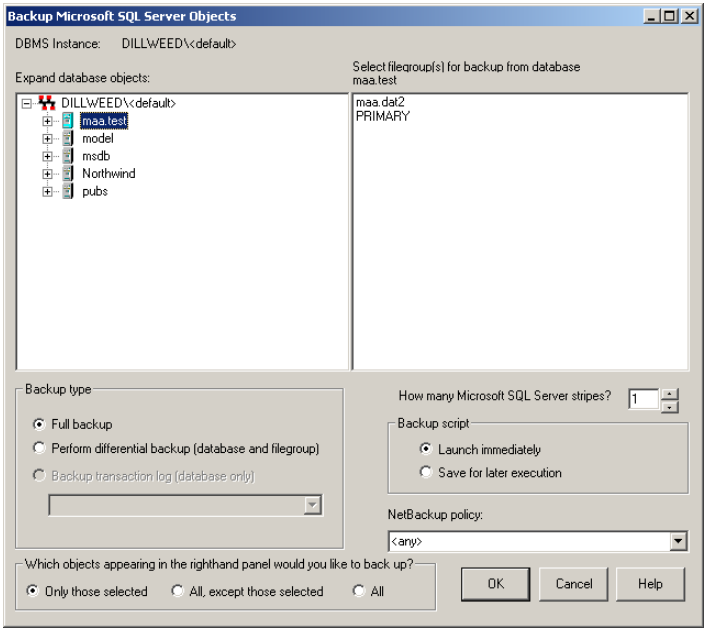
3. In the left pane, expand the instance name.

   The database names are displayed below the instance name.



4. In the left pane, select a database whose files you would like to back up.

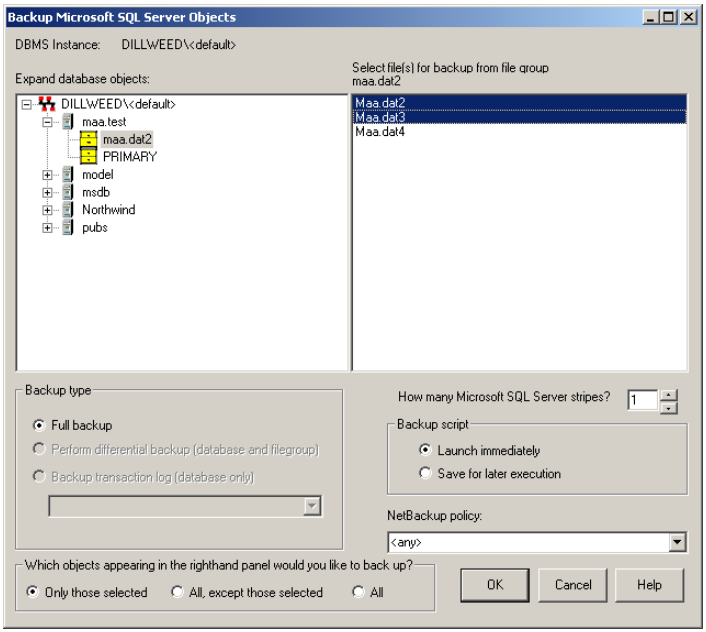
In the right pane, the filegroups belonging to the selected database are displayed.



5. In the left pane, expand the icon of the database containing the files that you want to backup, then select the filegroup that contains the files you want to back up.

6. In the right pane, select one or more files that you want to back up.

Shift + click or Ctrl + click to select more than one file.



7. Click **OK**.

8. When prompted to start the backup, click **Yes**.

NetBackup Restore Operations

Browsing Backup Images

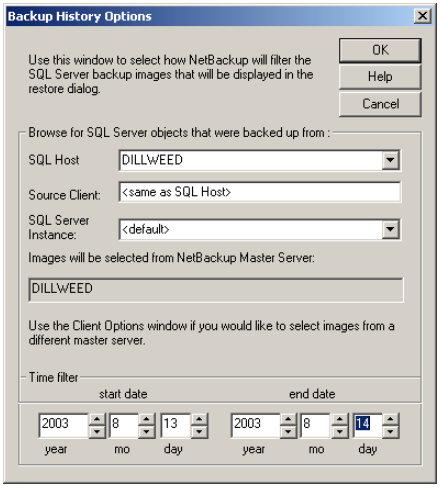
This section describes how to browse for available backup images. Once you have displayed the desired backup images, then follow the instructions for restoring a specific SQL object.

▼ To browse for backup images

1. To change the host and instance you wish to access, refer to "Setting the Database Login Parameters" on page 63.

2. Choose **Actions** > **Restore**.

   The Backup History Options dialog is displayed.



3. Select the **SQL Host** and **SQL Server Instance** whose backup images you would like to browse. (Refer to "Backup History Options" on page 222 for information on the **Source Client** setting, if necessary.)

   These drop-down list boxes can also be manually edited if the host and instance that you want do not appear in the drop-down list boxes. You can designate the default instance either by setting the **SQL Server Instance** box to **<default>** or to empty (not spaces).

4. Choose the date range to search.

5. Click **OK**.

   The Restore Microsoft SQL Server objects dialog is displayed.

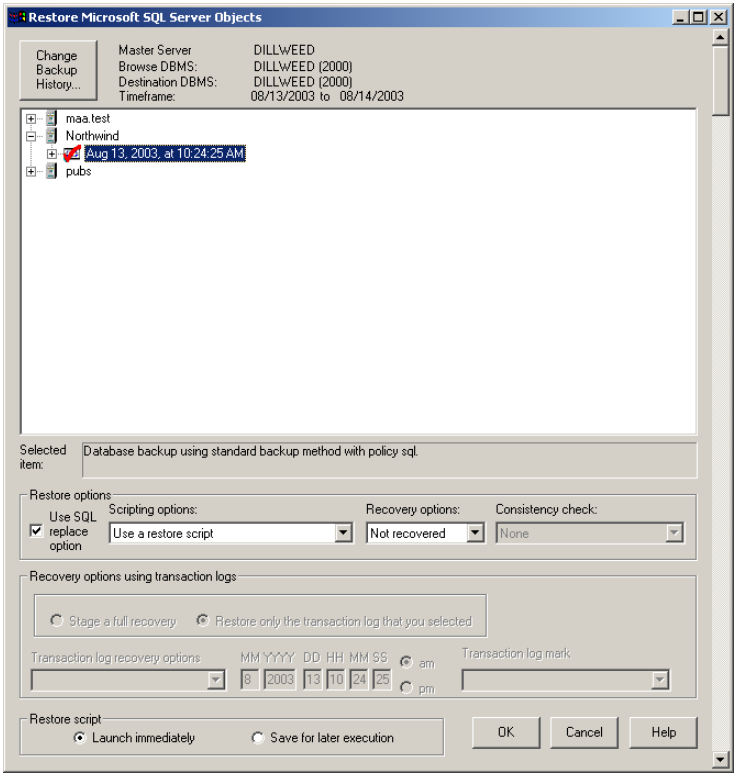
6. Continue with the instructions for restoring the desired database object:

Restoring a Database Backup

▼ To restore a database

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 75.

2. Select the database image that you wish to restore.



See "Restore Microsoft SQL Server Objects" on page 224 for a description of the available restore options.

3. Click **OK**.

4. When prompted to start the restore, click **Yes**.

Restoring a Database Differential Backup

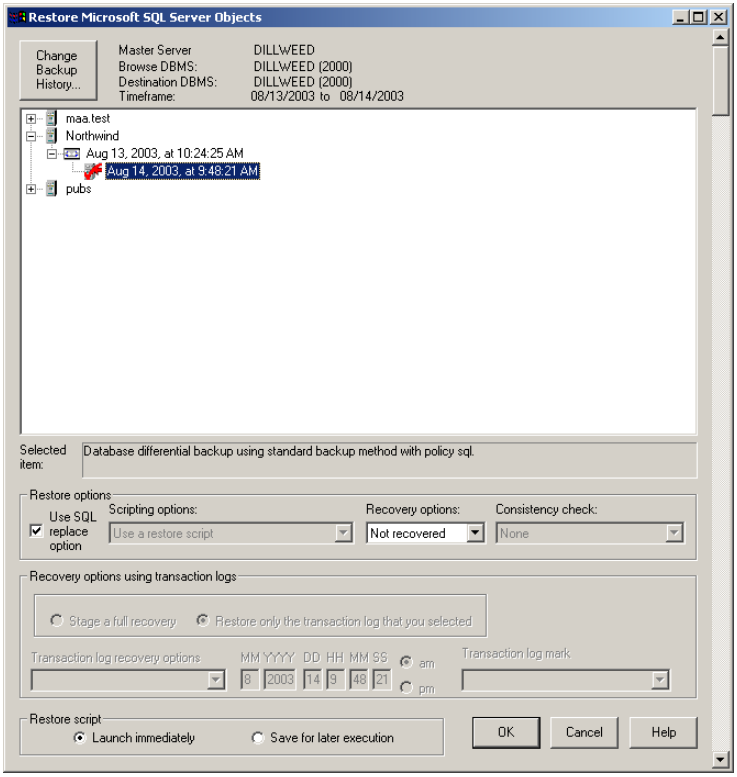
▼ To restore a database differential

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 75.

2. Click the "+" next to the database image whose differential you want to restore.

3. Select the database differential image that you wish to restore.

   See "Restore Microsoft SQL Server Objects" on page 224 for a description of the available restore options.



4. Click **OK**.

5. When prompted to start the restore, click **Yes**.

Restoring a Filegroup Backup

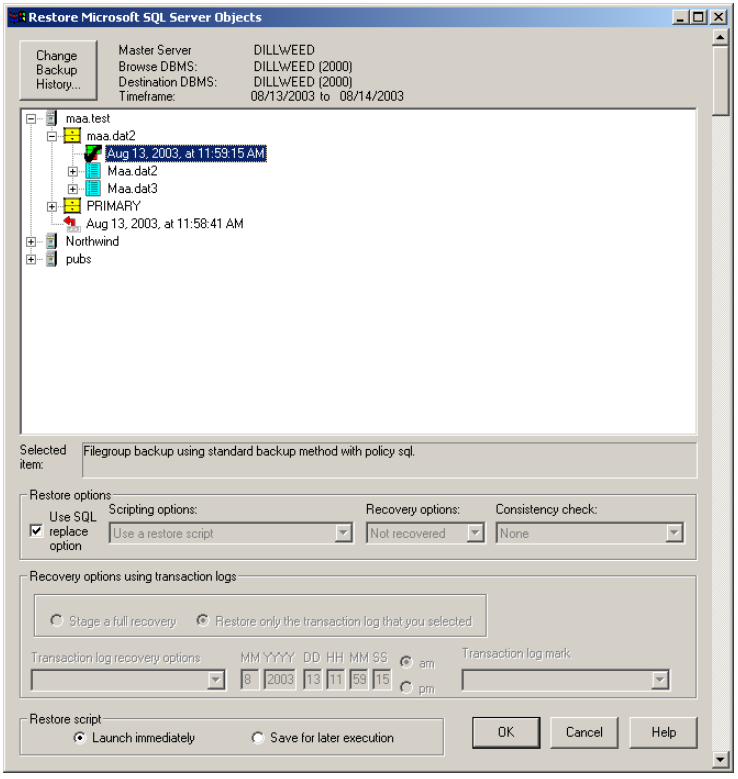
▼ To restore a filegroup

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 75.

2. Click the "+" next to the filegroup icon from which you would like to restore a filegroup backup image.

3. Select the filegroup image that you wish to restore.

   See "Restore Microsoft SQL Server Objects" on page 224 for a description of the available restore options.



4. Click **OK**.

5. When prompted to start the restore, click **Yes**.

Restoring a Filegroup Differential Backup

▼ To restore a filegroup differential

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 75.

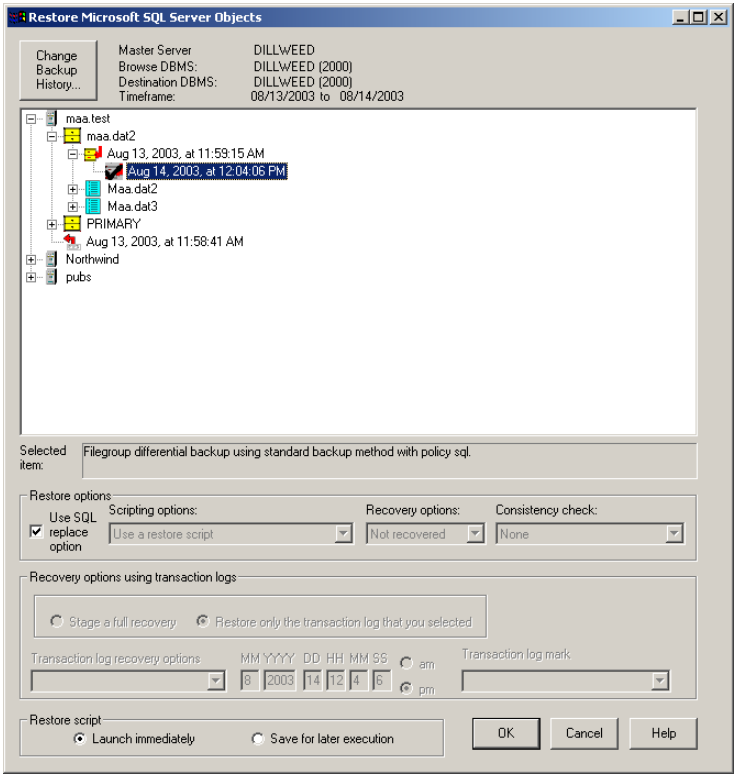
Caution If you attempt to restore a single differential backup without first restoring the preceding database backup file, SQL Server will halt the load process with an error such as 4305 or 4306. If you plan to restore a single differential, then you are responsible for first restoring the database backup file. You can avoid this problem by backing up the entire sequence of transaction logs, the differential backup, and the backup file to the same NetBackup Server and restoring the entire sequence of backup objects as described in "Restoring a Transaction Log Image Within a Staged Recovery" on page 84.

2. Click the "+" next to the filegroup icon from which you would like to restore a filegroup differential backup image.

3. Click the "+" next to the filegroup backup image from which you would like to restore the filegroup differential image.

4. Select the filegroup differential image that you want to restore.

See "Restore Microsoft SQL Server Objects" on page 224 for a description of the available restore options.



5. Click **OK**.

6. When prompted to start the restore, click **Yes**.

Restoring a File

▼ To restore a file

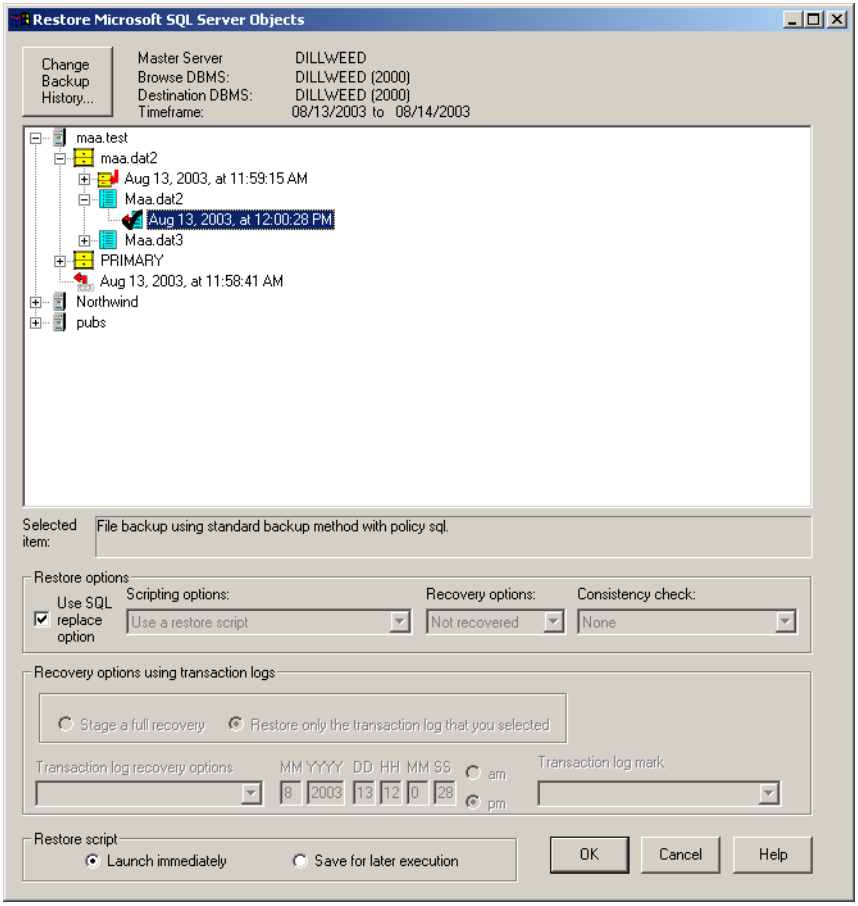
1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 75.

2. Click the "+" next to the filegroup icon which contains the file you would like to restore.

3. Click the "+" next to the file icon from which you would like to restore a file backup image.

4. Select the file image that you want to restore.

   See "Restore Microsoft SQL Server Objects" on page 224 for a description of the available options.



5. Click **OK**.

6. When prompted to start the restore, click **Yes**.

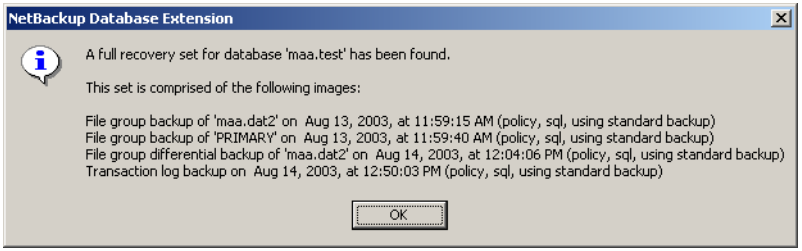
Restoring a Transaction Log Image Without Staging a Full Recovery

▼ To restore a transaction log without staging a full recovery

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 75.

2. Select the transaction log image that you want to restore.

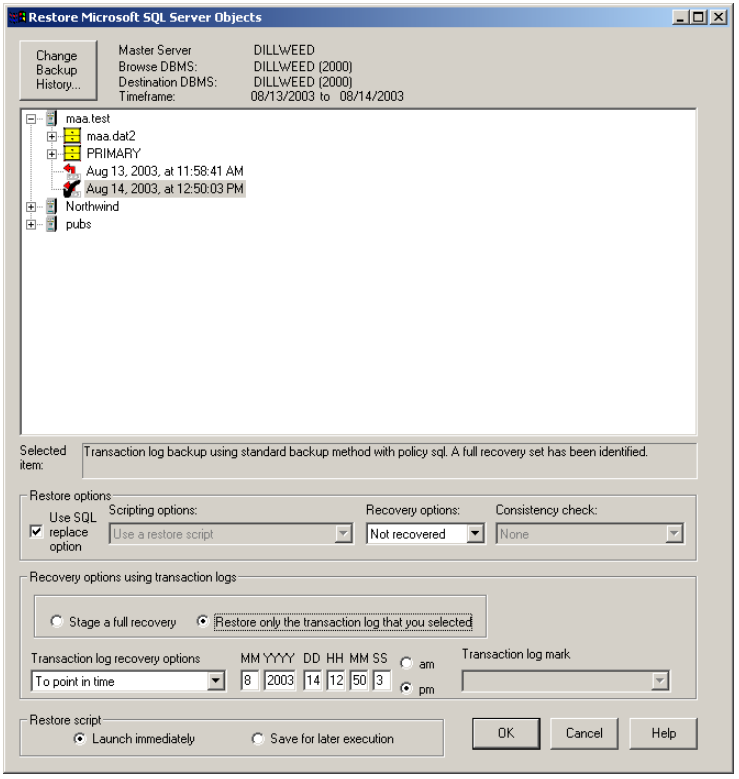
   If NetBackup can discover a set of images which include the selected transaction log image, then a message box is displayed.



Note This message will only appear if NetBackup can find a set of images that constitutes a full recovery for the selected database. If the message does not appear, then NetBackup is not able to create a staged restore script for you.

3. To close the message box, click **OK**.

4. Select **Restore only the transaction log that you selected**.



5. Click **OK**.

6. When prompted to start the restore, click **Yes**.

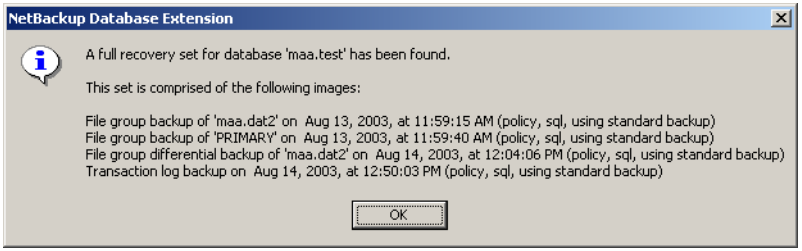
Restoring a Transaction Log Image Within a Staged Recovery

▼ To restore a transaction log within a staged recovery

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 75.

2. Select the transaction log image that you want to restore.

   If NetBackup can discover a set of images which include the selected transaction log image, then a message box is displayed.

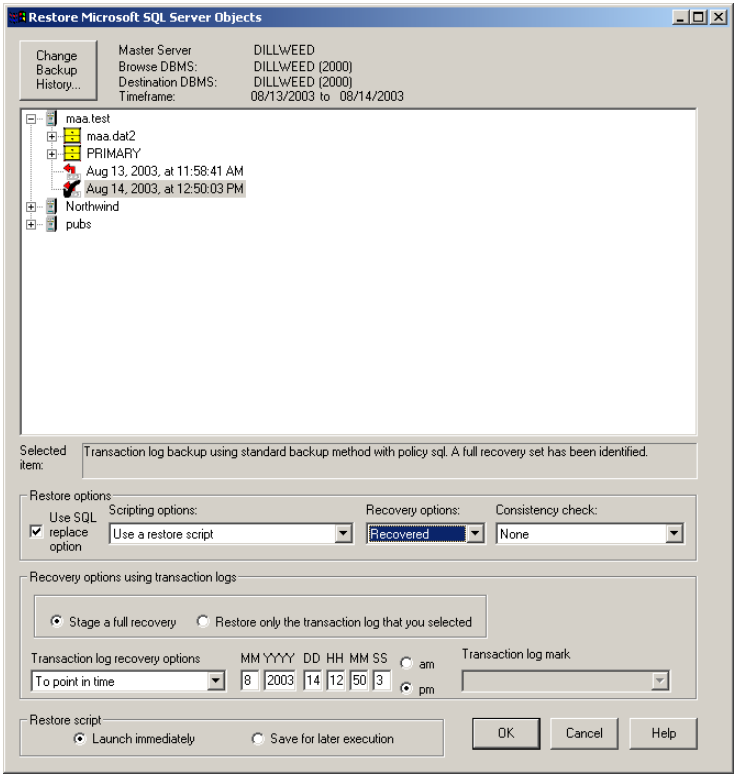


Note This message will only appear if NetBackup can find a set of images that constitutes a full recovery for the selected database. If the message does not appear, then NetBackup is not able to create a staged restore script for you.

3. To close the message box, click **OK**.

   The Restore Microsoft SQL Server Objects dialog is displayed.

4. Select **Stage a full recovery**.



5. Click **OK**.

6. When prompted to start the restore, click **Yes**.

Performing a Database Move

The following instructions show you how to perform a database move. A database move allows you to use a full set of backup images to copy an existing database to a new database having a different name.

Database move operations can only be carried out when your selection includes a database image. This can occur either when you directly select the database backup image, or when NetBackup finds a recovery set which contains a database backup image.

Tip When you are preparing your database design, map logically related database components such as tables, indexes, keys, etc., to the same filegroup. This will enable you to use the partial database restore procedure, which follows, to recover these logically related components as a single unit.

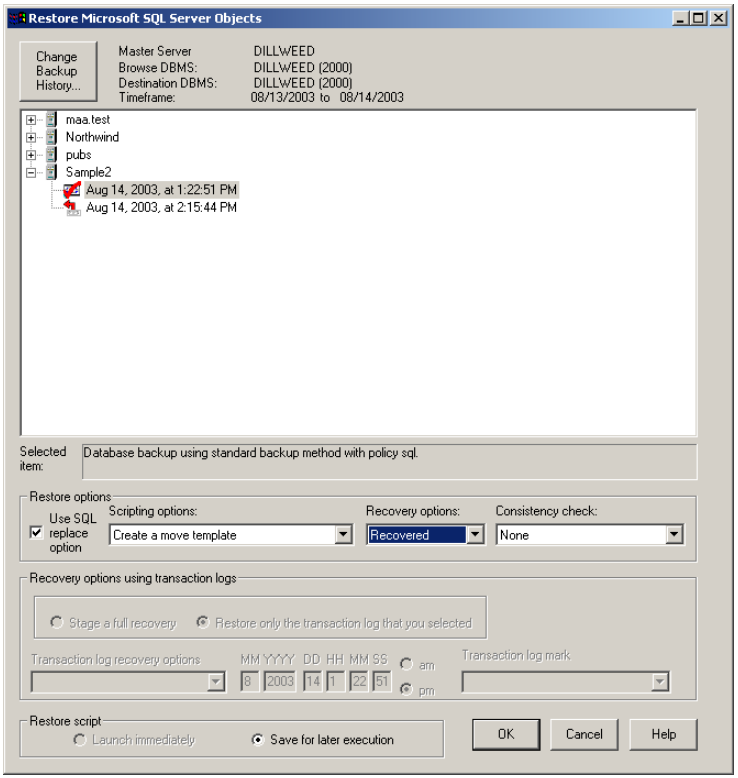
In the following example the move script is created upon selection of a database backup image.

▼ To perform a database move

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 75.

2. Select the database backup image that you want to restore.

3. From the **Scripting options** list, select **Create a move template**.

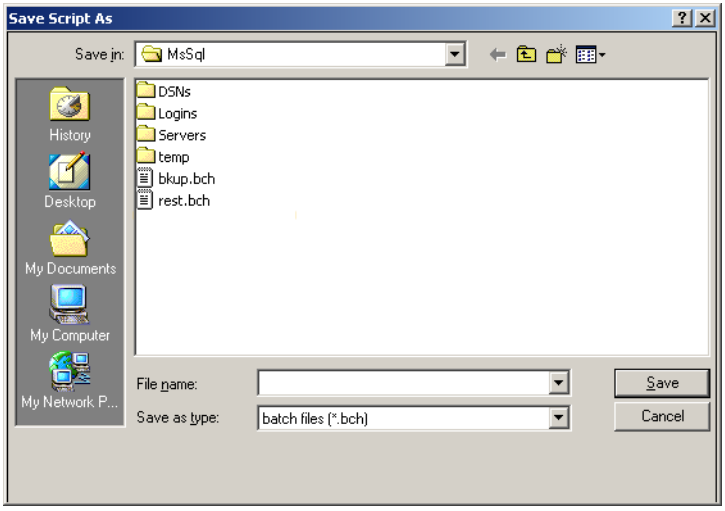


Note When you choose to create a move or partial database restore script, the capability to perform an immediate launch is disabled because you will need to edit the script in order to specify certain destination parameters.

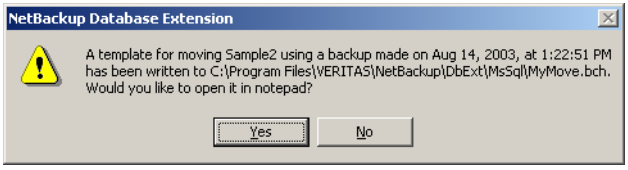
4. Click **OK**.

   The Save Script As dialog is displayed.

5. Type in the name of a file to which you want NetBackup to write the move template and click **Save**.



A message box is displayed.



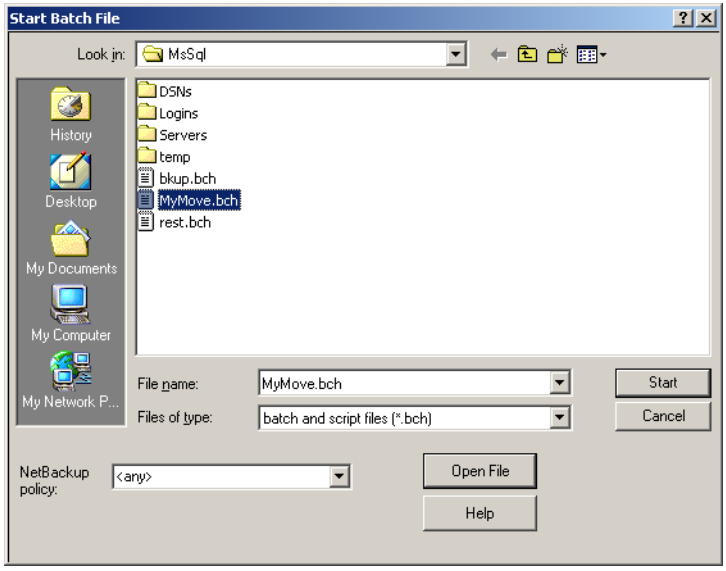
6. Click **Yes** to open the move template in Notepad.

   The contents of the sample move template are displayed below. It contains instructions on how you can modify this template in order to perform a full database restore to a database with a different name.

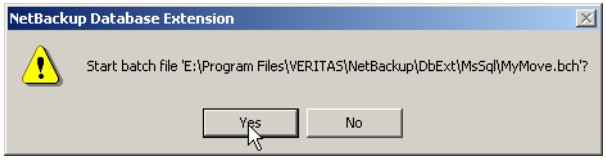
```
#
#  This is a template for the database MOVE command.
#
OPERATION RESTORE
OBJECTTYPE DATABASE
#
#  Replace the database name in the following line with the name of the database that you
#  want to move to. Also remove the hash mark <#> which precedes the keyword <DATABASE>.
#
#DATABASE "Sample2"
#
#  Replace the file path <C:\Program Files\Microsoft SQL Server\MSSQL\data\Sample2.mdf>
#  with a new file path. Also remove the hash mark <#> which precedes the keyword <TO>.
#  The target of the MOVE keyword must be "Sample2_Prm".
MOVE  "Sample2_Prm"
#TO  "C:\Program Files\Microsoft SQL Server\MSSQL\data\Sample2.mdf"
#
#
#  Replace the file path <c:\db\Sample2_FG_Dat2.ndf>
#  with a new file path. Also remove the hash mark <#> which precedes the keyword <TO>.
#  The target of the MOVE keyword must be "Sample2_FG1_Dat1".
MOVE  "Sample2_FG1_Dat1"
#TO  "c:\db\Sample2_FG_Dat2.ndf"
#
#
#  Replace the file path <c:\db\Sample2_FG_Dat3.ndf>
#  with a new file path. Also remove the hash mark <#> which precedes the keyword <TO>.
#  The target of the MOVE keyword must be "Sample2_FG1_Dat2".
MOVE  "Sample2_FG1_Dat2"
#TO  "c:\db\Sample2_FG_Dat3.ndf"
#
#
#  Replace the file path <c:\db\Sample2_FG_Dat4.ndf>
#  with a new file path. Also remove the hash mark <#> which precedes the keyword <TO>.
#  The target of the MOVE keyword must be "Sample2_FG1_Dat3".
MOVE  "Sample2_FG1_Dat3"
#TO  "c:\db\Sample2_FG_Dat4.ndf"
#
#
#  Replace the file path <C:\Program Files\Microsoft SQL Server\MSSQL\data\Sample2.ldf>
#  with a new file path. Also remove the hash mark <#> which precedes the keyword <TO>.
#  The target of the MOVE keyword must be "Maa.log".
MOVE  "Maa.log"
#TO  "C:\Program Files\Microsoft SQL Server\MSSQL\data\Sample2.ldf"
#
NBIMAGE "dillweed.MSSQL7.DILLWEED.db.Sample2.~.0.001of001.20030814132251..C"
SQLHOST "DILLWEED"
NBSERVER "DILLWEED"
BROWSECLIENT "DILLWEED"
MAXTRANSFERSIZE 0
BLOCKSIZE 0
RESTOREOPTION REPLACE
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```

7. After you have modified the script template created in step 6, choose **Actions** > **Batch Files**.

8. Select the batch file that you created, and select **Start**.



9. Click **Yes** to launch the restore.



Performing a Partial Database Restore

The following instructions show you how to perform a partial database restore. A partial database restore lets you select individual filegroup components of a database and restore them to a new database having a different name.

Tip When you are preparing your database design, map logically related database components such as tables, indexes, keys, etc., to the same filegroup. This will enable you to use the partial database restore procedure, which follows, to recover these logically related components as a single unit.

Note Partial database restore is only available if SQL Server 2000 is installed.

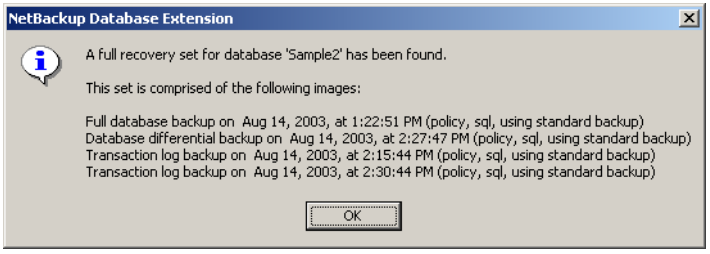
In the following example the partial database restore template is created upon selection of a transaction log backup image which belongs to a full recovery set that NetBackup detects.

▼ To perform a partial database restore

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 75.

2. Select the transaction log image that you want to restore.

   If NetBackup can discover a set of images which include the selected transaction log image, then a message box is displayed.

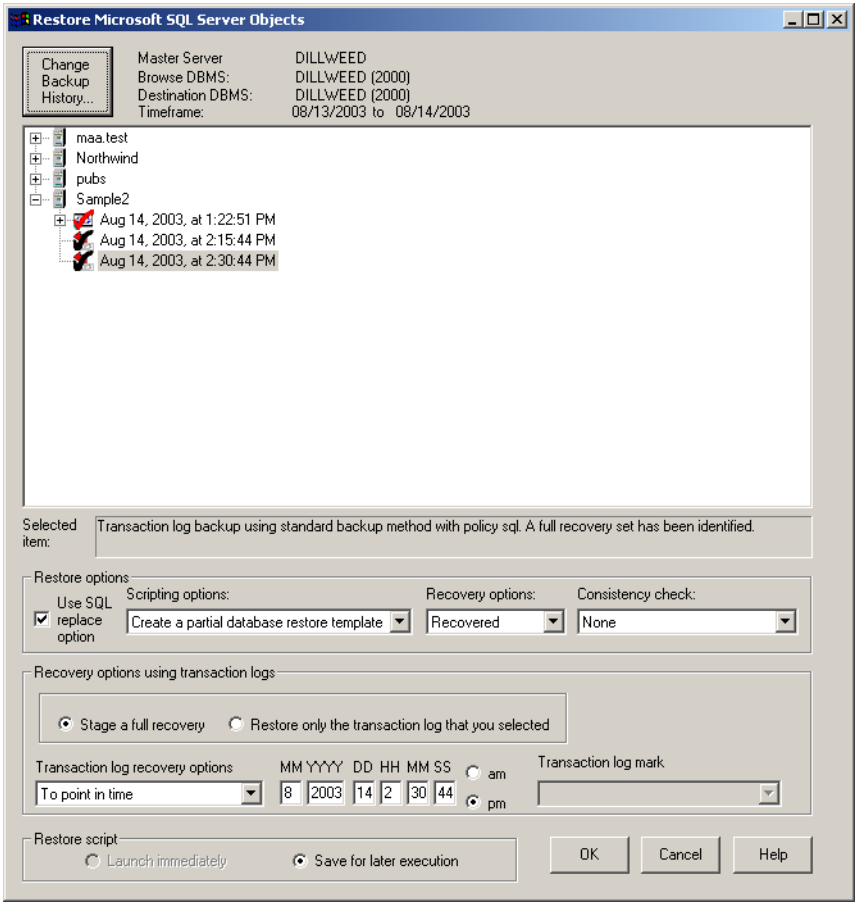


Note This message box will only appear if NetBackup can find a set of images that constitutes a full recovery for the selected database. If the message box does not appear, then NetBackup is not able to create a staged restore script for you. In order to create either a partial database restore template or database move template, there must be a full recovery set available for NetBackup to discover and the set must include a full database backup.

3. Click **OK**.

4. From the **Scripting options** list, select **Create a partial database restore template**.

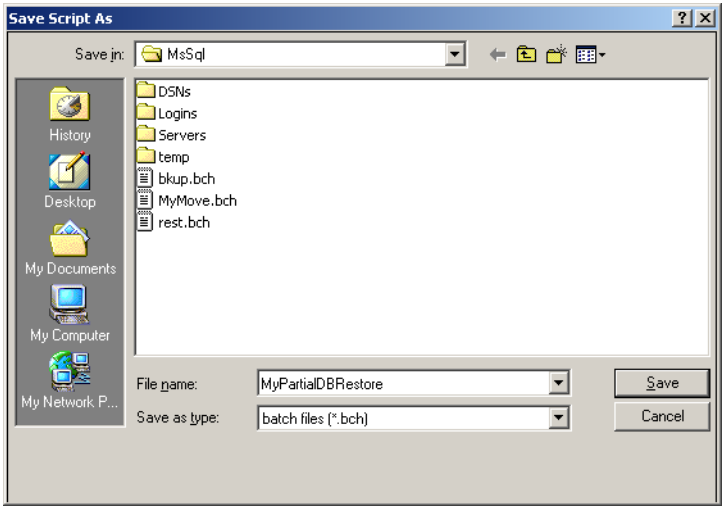
Note When you choose to create a move or partial database restore script, the capability to perform an immediate launch is disabled because you will need to edit the script in order to specify certain destination parameters.



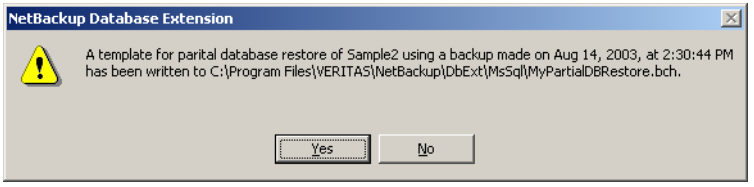
5. Click **OK**.

The Save Script As dialog is displayed.

6. Type the name of a file to which you want NetBackup to write the partial database restore template and click **Save**.



The following message box is displayed.



7. Click **Yes** to open the partial database restore template in Notepad.

   The contents of the sample partial database restore template are displayed below. It contains instructions on how you can modify this template in order to perform a partial database restore.

```
#
#  This is a template for PARTIAL DATABASE RESTORE.
#  It requires Microsoft SQL Server 2000.
#
OPERATION RESTORE
OBJECTTYPE DATABASE
#
#  Replace the database name in the following line with the name of the database that you
#  want as the target of the partial restore. Also remove the hash mark <#> which precedes
#  the keyword <DATABASE>.
#DATABASE "Sample2"
#
#  If you wish to restore file <C:\Program Files\Microsoft SQL Server\MSSQL\data\Sample2.mdf>,
```

```
#  then replace this path with a new file path. Also remove the hash marks <#>
#  which precede the keywords <MOVE> and <TO>.
#  The target of the MOVE keyword must be "Sample2_Prm".
#MOVE  "Sample2_Prm"
#TO  "C:\Program Files\Microsoft SQL Server\MSSQL\data\Sample2.mdf"
#
#
#  If you wish to restore file <c:\db\Sample2_FG_Dat2.ndf>,
#  then replace this path with a new file path. Also remove the hash marks <#>
#  which precede the keywords <MOVE> and <TO>.
#  The target of the MOVE keyword must be "Sample2_FG1_Dat1".
#MOVE  "Sample2_FG1_Dat1"
#TO  "c:\db\Sample2_FG_Dat2.ndf"
#
#
#  If you wish to restore file <c:\db\Sample2_FG_Dat3.ndf>,
#  then replace this path with a new file path. Also remove the hash marks <#>
#  which precede the keywords <MOVE> and <TO>.
#  The target of the MOVE keyword must be "Sample2_FG1_Dat2".
#MOVE  "Sample2_FG1_Dat2"
#TO  "c:\db\Sample2_FG_Dat3.ndf"
#
#
#  If you wish to restore file <c:\db\Sample2_FG_Dat4.ndf>,
#  then replace this path with a new file path. Also remove the hash marks <#>
#  which precede the keywords <MOVE> and <TO>.
#  The target of the MOVE keyword must be "Sample2_FG1_Dat3".
#MOVE  "Sample2_FG1_Dat3"
#TO  "c:\db\Sample2_FG_Dat4.ndf"
#
#
#  If you wish to restore file <C:\Program Files\Microsoft SQL Server\MSSQL\data\Sample2.ldf>,
#  then replace this path with a new file path. Also remove the hash marks <#>
#  which precede the keywords <MOVE> and <TO>.
#  The target of the MOVE keyword must be "Maa.log".
#MOVE  "Maa.log"
#TO  "C:\Program Files\Microsoft SQL Server\MSSQL\data\Sample2.ldf"
#
NBIMAGE "dillweed.MSSQL7.DILLWEED.db.Sample2.~.0.001of001.20030814132251..C"
SQLHOST "DILLWEED"
NBSERVER "DILLWEED"
BROWSECLIENT "DILLWEED"
MAXTRANSFERSIZE 0
BLOCKSIZE 0
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE DATABASE
DUMPOPTION INCREMENTAL
#
#  Replace the database name in the following line with the name of the database that you
#  want as the target of the partial restore. Also remove the hash mark <#> which precedes
```


```
#  the keyword <DATABASE>.
#DATABASE "Sample2"
NBIMAGE "dillweed.MSSQL7.DILLWEED.inc.Sample2.~.0.001of001.20030814142747..C"
SQLHOST "DILLWEED"
NBSERVER "DILLWEED"
BROWSECLIENT "DILLWEED"
MAXTRANSFERSIZE 0
BLOCKSIZE 0
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE TRXLOG
#
#  Replace the database name in the following line with the name of the database that you
#  want as the target of the partial restore. Also remove the hash mark <#> which precedes
#  the keyword <DATABASE>.
#DATABASE "Sample2"
NBIMAGE "dillweed.MSSQL7.DILLWEED.trx.Sample2.~.0.001of001.20030814141544..C"
SQLHOST "DILLWEED"
NBSERVER "DILLWEED"
BROWSECLIENT "DILLWEED"
MAXTRANSFERSIZE 0
BLOCKSIZE 0
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE TRXLOG
#
#  Replace the database name in the following line with the name of the database that you
#  want as the target of the partial restore. Also remove the hash mark <#> which precedes
#  the keyword <DATABASE>.
#DATABASE "Sample2"
NBIMAGE "dillweed.MSSQL7.DILLWEED.trx.Sample2.~.0.001of001.20030814143044..C"
SQLHOST "DILLWEED"
NBSERVER "DILLWEED"
BROWSECLIENT "DILLWEED"
MAXTRANSFERSIZE 0
BLOCKSIZE 0
RESTOREOPTION REPLACE
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```

8. After you have modified the script template created in step 7, choose **Actions** > **Batch Files**.

9. Select the batch file that you created, and click **Start**.

10. Click **Yes** to launch the partial database restore.

Redirecting a Restore to a Different Host

Multiple NetBackup database clients may be configured to use a single NetBackup master server for backing up SQL Server databases. With this configuration, you can back up SQL Server database objects from one client and restore them to another.

▼ To perform a redirected restore of a SQL Server object backed up on HostA to HostB

1. Establish permission settings on the master server. Do one of the following:

   - Create a file called *install_path*\NetBackup\db\altnames\No.Restrictions, to allow unrestricted redirected restore privileges

   OR

   - Create a file called *install_path*\NetBackup\db\altnames\<HostB>, to allow HostB to restore HostA's data. Note that this allows the client named HostB to access HostA's data on the master as well as any other client's data that was backed up on the master.

2. Choose **Options** > **Set DBMS login parameters**.

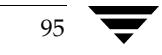
3. Log onto the target instance on HostB.

4. Choose **Options** > **Set NetBackup Client Options** and select the current master server to be a master server which is common to both HostA and HostB.

5. Choose **Actions** > **Restore**.

6. From the Backup History Options dialog, select HostA as the SQL host and select the instance on HostA from which the target objects were backed up.

When the restore window is displayed it will contain a history of objects that were backed up from HostA. You can then select those objects and restore them to HostB.



Redirecting a Database to a Different Location on a Different Host

The database move redirects the restore of a database to a different location. The new location may be a different instance on the same host, a different host, or a different file path. The move operation also allows you to restore the database under a different name than the original one. The move operation is not available for images created using SQL Server 6.5, nor can SQL Server 7.0 or 2000 images be moved to a SQL Server 6.5 installation.

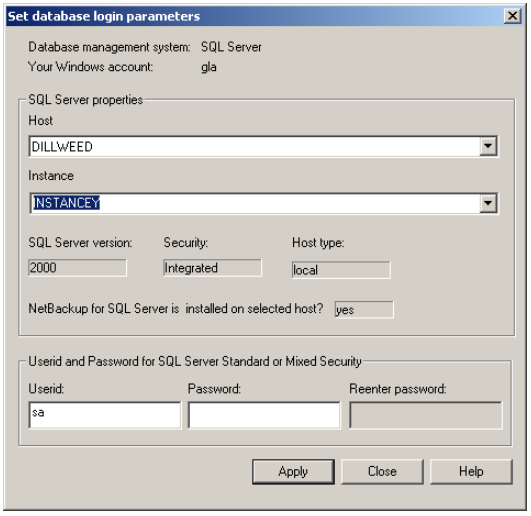
Note The destination host\instance of a move or restore operation is the one that you log into as indicated by the Set database login parameters dialog. The source (or browse) host\instance for move or restore operations is designated using the Backup History Options dialog, which comes up automatically when you choose **Actions** > **Restore**.

Note If you have backup images created with a pre-4.5 version of NetBackup, see "Creating a Move Template for Images Created with Pre-4.5 Versions of NetBackup" on page 100.

▼ To redirect a database to another location on a different host

1. The master server which backed up the database you wish to restore must appear in the server list of the destination host. If it is not, see "Selecting a Master Server" on page 100.
2. Choose **Options > Set database login parameters**.
3. From the **Host** list, select the host you wish to restore to.
4. From the **Instance** list, select the desired instance.

To choose the default instance, either select **<default>** or set the **Instance** to empty (not spaces).

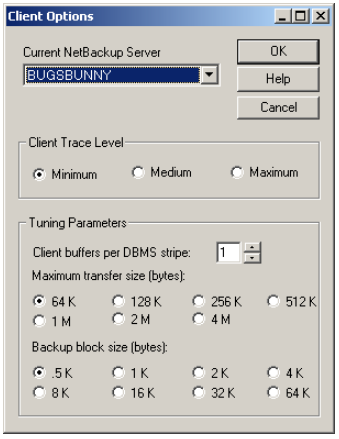


5. Click **Apply** and **Close**.

6. Choose **Options** > **Set NetBackup Client Options**.

   The Client Options dialog is displayed.

7. From the **Current Netbackup Server** list, select the NetBackup Server which contains the SQL Server backup images that you want to restore on the destination host.



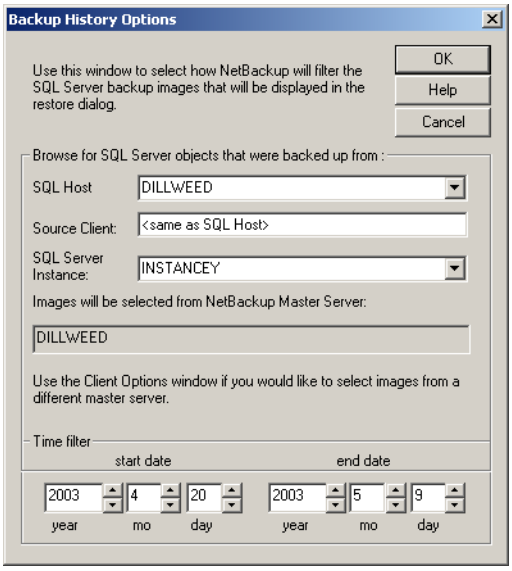
8. Click **OK**.

9. Choose **Actions** > **Restore**.

   The Backup History Options dialog is displayed.

10. In the **SQL Host** list, select the host which has the database you wish to restore.

    (Refer to "Backup History Options" on page 222 for information on the **Source Client** setting, if necessary.)

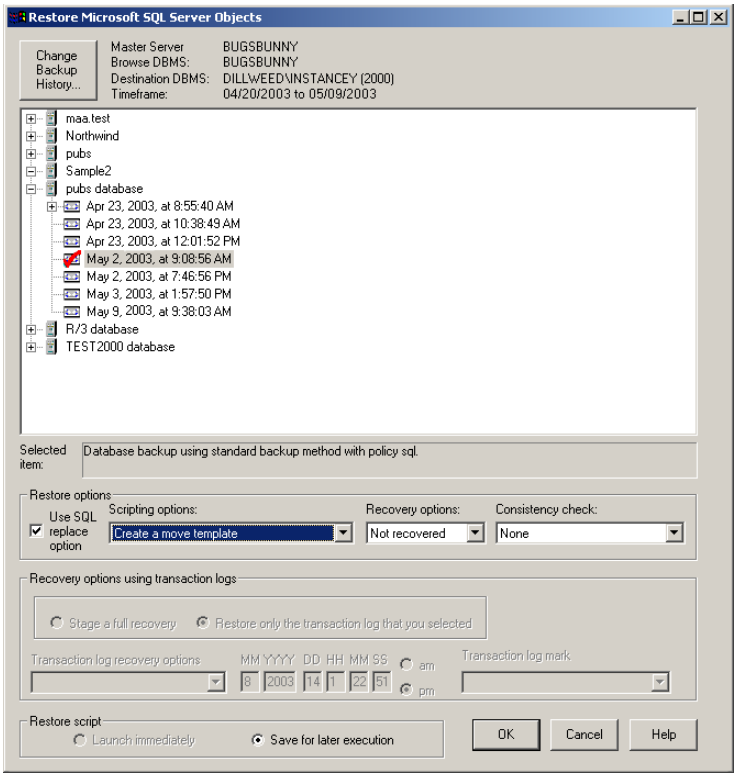


11. From **SQL Server Instance** list, select the instance name of the source.

12. Click **OK**.

13. Browse for the database that you want to move.

14. From the **Scripting options** list, select **Create a move template**.



15. Click **OK**.

    A Save As dialog is displayed.

16. Enter a batch file name, and click **Save**.

17. When the NetBackup Database Extension message box is displayed, click **Yes** to open the move template that was created.

18. Edit this template to designate the name that you would like to use for the destination database as well as the file paths that you would like to use for each of the database files.

19. After you have edited the template, you can launch it by choosing **Actions** > **Batch Files**.

Creating a Move Template for Images Created with Pre-4.5 Versions of NetBackup

In order to create a move template, NetBackup for SQL Server needs to know the file structure of the source database. For NetBackup for SQL Server images created with version 4.5 or greater, this information is stored in the NetBackup catalog. However, in order to create a move script using an image created in an earlier version of NetBackup, you must log into the same SQL Server instance on the same host machine from which the image was backed up. In addition, for pre-4.5 images, you must not have changed the file structure of the database by adding or deleting files or filegroups.

You do not necessarily need to use a move script when you restore a database to a different host or database instance. However, if you do not use a move script, then you may get SQL Server error messages such as 3156 or 3157, which indicate that the original source file path is not available on the destination machine. This may occur because the path is already in use for a different database or because the path does not exist. You can generally resolve these errors by creating a move script and designating new file paths, which do exist and are unused, on the destination host.

Selecting a Master Server

When you perform a move or restore, the images must be available on the host machine which is acting as the NetBackup master server for the destination host. If this master server is contained in the destination's server list, then you can select the current server by choosing **Options** > **Set NetBackup Client Options**.

If the master server is not in the server list of the destination host — perhaps, because the server machine is remote or has access limitations — then you must duplicate the images onto removable media (with a unique id), transport that media to the master server used by the destination host, and import the images to the master server used by the destination host. After the images have been imported, continue with the following instructions.

Using Batch Files

NetBackup for SQL Server uses scripts called batch files for initiating backup and restore operations. A batch file uses the `.bch` extension and is typically executed from the *`install_path`*`\dbext\mssql\` directory.

Overview of Batch Files

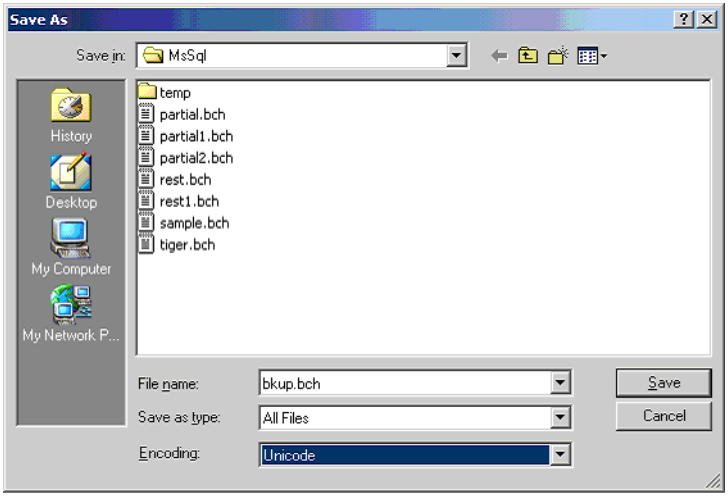
You will have to create a batch file if you start up operations with:

- the Start Batch File dialog in NetBackup for SQL Server
- the `dbbackex` command line
- the NetBackup Server Scheduler
- the NetBackup Administration Console for Windows or for UNIX

You can easily create a fully functional batch file by selecting the **Save for Later Execution** radio button from any of the backup or restore dialogs and then clicking **OK**. The following sections, however, provide guidelines to help you to understand the contents of a batch file and how to modify or create one on your own.

Text Format Used in Batch Files

Batch files created by the SQL Server Agent GUI are Unicode text. This permits inclusion of object names and image names that include non-English characters. If you create a batch file manually, use notepad or some other tool that generates plain text. If your batch file contains any non-English characters save it as Unicode as shown below.



Guidelines for Creating Batch Files

- Give the file a unique name with the extension `.bch`.
- Place the batch file in the *`install_path`*`\NetBackup\dbext\MSSQL\` folder.

By following these guidelines, you will benefit as follows:

- When you open the Start Batch File dialog from the NetBackup Database Extension GUI, icons for your batch files are displayed immediately in the display window.
- When you launch a job through the `dbbackex` command line interface, specify only the file name (not the full path name) on the command line.
- When you specify a file name for the file list of an MS-SQL-Server policy schedule, you only need to specify the file name (not the full path name).

Notes

There are several points to remember about batch files.

- A batch file consists of a series of operations (backups and restores) which by default are run in sequence.
- Each operation consists of a series of <*keyword value*> pairs, which completely define the total operation.
- The keyword is not case sensitive but the value is. Generally, you will be safe if you code both the keyword and value in uppercase, with the exception that, if you use the NBIMAGE keyword option, then the value must be specified exactly as it is stored by NetBackup Server.
- Operations are not nested.
- With the exception of the BATCHSIZE and GROUPSIZE parameters, <*keyword value*> pairs are not global. If you use BATCHSIZE or GROUPSIZE, then it must appear only once in your batch file and it must appear in the first operation.
- Within an operation, the <*keyword value*> pairs may appear in any order except that each operation must be terminated by `ENDOPER TRUE`.
- You can include comment lines in your batch file by placing a hash mark ('#') in the first column.

Keywords and Values Used in Batch Files

This section describes the keywords and values used in a batch file.

Keywords and Values Used in Batch Files

Keyword	Values	Required?	Default	Description
ALTCLIENT (Same as BROWSECLIENT)	string	no	None	Restores images from a host other than the local host.
BACKUPMODEL	BACKUPMODEL_CONVENTIONAL, BACKUPMODEL_SNAPSHOT	no	BACKUPMODEL_CONVENTIONAL	Valid only for restore. Indicates whether the backup was originated from a snapshot method
BATCHSIZE	integer	no	1	Number of operations to start up simultaneously. Applies to all of the operations in the batch file. Must appear before the end of the first operation. Range is 1-10.
BLOCKSIZE	integer	no	0	Applicable for backup operations only. Block size is calculated as 512 bytes * $2^{BLOCKSIZE}$. Range is 1-7.
BROWSECLIENT (Same as ALTCLIENT)	string	no	None	Restores images from a host other than the local host.
BUFFERS	integer	no	1	Number of buffers per stripe. Range is 1-32.
CLOAKEDBACKUP	TRUE or FALSE	no	FALSE	Applies only to Advanced Client database backups. If TRUE, the backup will not be recorded by SQL Server and therefore cannot be the basis for any subsequent differential backup. See "Using a Cloaked Database Snapshot to Impact How Differentials are Based" on page 169.
CONSISTENCYCHECK	FULLINCLUDING INDICES, FULLEXCLUDING INDICES, PHYSICALCHECK ONLY, CHECKCATALOG	no	none	Performs the specified consistency check after the restore has been completed.

Keywords and Values Used in Batch Files

Keyword	Values	Required?	Default	Description
DATABASE	string	yes	none	Name of database. For backup operations, specify value **$ALL** to designate all databases (except for tempdb.)
DBMS	MSSQL	no	MSSQL	You can specify MSSQL only.
DSN	string	no	saved from GUI user session	ODBC data source name

Note Although the DSN keyword is supported, its usage is deprecated. Specify the SQL Server host and instance using the SQLHOST and SQLINSTANCE parameters.

DUMPOPTION	INCREMENTAL	no	none	Specifies INCREMENTAL restoring from an incremental backup.
ENDOPER	TRUE	yes	none	Terminates each operation specified in the batch file.
EXCLUDE	string	no	none	Name of a database to exclude when DATABASE $ALL is specified in a batch operation

Note EXCLUDE can be used in a batch file only if DATABASE $ALL is used.

GROUPSIZE	integer between 1 and 32	no	none	The number of databases that will be snapped as a single SQL Server backup image. See "Using NetBackup for SQL Server with Advanced Client" on page 159.
INHIBITALTBUFFERMETHOD	TRUE, FALSE	no	FALSE	Tells NetBackup whether to consider the candidacy of alternate buffer method.
MAXTRANSFERSIZE	integer	no	0	Maximum transfer size is calculated as 64 kilobytes bytes * 2MAXTRANSFERSIZE. Range is 1-6.
MOVE	file group	no	none	Specifies a filegroup name. Used for restore types PARTIAL and MOVE.
NBIMAGE	string	yes*	none	Specifies a NetBackup image for the restore operations. See note below. *Required for restore operations.

Keywords and Values Used in Batch Files

Keyword	Values	Required?	Default	Description
NBSCHED	string	no	none	If the NetBackup policy has several *Application Backup Policy* schedules, use NBSCHED to select amongst them.
NBSERVER	string	no	none	Specifies which master server to use for the backup or restore operation.

Note If NBSERVER is not specified in a batch file operation, the master server defaults to the name specified at HKEY_CURRENT_USER\Software\VERITAS\NetBackup\NetBackup for Microsoft SQL Server\DEFAULT_SQL_NB_MASTER_SERVER.

NUMBUFS				Refer to BUFFERS.
OBJECTNAME	string	yes*	none	Specifies a file or filegroup name for file/filegroup backups and restores, * If OBJECTTYPE= FILE or FILEGROUP.
OBJECTTYPE	DATABASE, TRXLOG, FILEGROUP, FILE	no	DATABASE	Specifies whether you are backing up or restoring a database, transaction log, filegroup, or file.
OBJTYPE				Refer to OBJECTTYPE.
OPERATION	BACKUP RESTORE	no	BACKUP	Type of operation, either backup or restore.
PASSWORD	string	no	null	Password for logging into SQL Server.
RECOVERED STATE	RECOVERED, STANDBY, NOTRECOVERED, TRUE, FALSE	no	RECOVERED	See the following note.

Note Applies to SQL Server 7.0 or 2000 restores. RECOVEREDSTATE means that the database should be restored to the recovered state. NOTRECOVERED means that it should remain in the loading state following the restore. STANDBY means that the database should be restored to standby state. If STANDBY is used, then the STANDBYPATH keyword is also required. TRUE and FALSE, when used as values for RECOVEREDSTATE, are synonyms for RECOVERED and NOTRECOVERED.

Keywords and Values Used in Batch Files

Keyword	Values	Required?	Default	Description
RESTOREBEFOREMARK	string	no	none	SQL Server 2000 restore option. Specify transaction log mark.
RESTOREBEFOREMARK AFTERTIME	string	no	none	SQL Server 2000 restore option. Specify transaction log mark.
RESTOREOPTION	REPLACE	no	none	Tells NetBackup to use the SQL Server replace option on a restore.
RESTORETOMARK	string	no	none	SQL Server 2000 restore option. Specify transaction log mark.
RESTORETOMARK AFTERTIME	string	no	none	SQL Server 2000 restore option. Specify transaction log mark.
RESTORETYPE	FULL, PARTIAL, MOVE	no	FULL	Full = Full database restore, Partial = Partial database restore, Move = Database move

Note RESTORETYPE is applicable only to RESTORE database operations. PARTIAL requires SQL Server 2000. If MOVE is used, then the batch file should contain a series of one or more
<MOVE><filegroup>
<TO><file path>
sequences. If PARTIAL is used, then a similar sequence is also required, but the sequence for PARTIAL must specify all of the filegroups in the database whose backup image is referenced by the NBIMAGE keyword.

SQLHOST	string	no	See note	Name of SQL Server host.
SQLINSTANCE	string	no	See note	Name of SQL Server instance

Note If SQLHOST is not specified in a batch file operation, then the SQL Server host is obtained from HKEY_CURRENT_USER\Software\VERITAS\NetBackup\NetBackup for Microsoft SQL Server\DEFAULT_SQL_HOST. If SQLINSTANCE is missing, then the default SQL Server instance is assumed for the SQL Host.

STANDBYPATH	string	no	none	Specify a fully- qualified file path to use for the standby redo log.

Keywords and Values Used in Batch Files

Keyword	Values	Required?	Default	Description
STOPAT	datetime string	no	none	Specifies point-in-time recovery of a transaction log. The datetime string is formatted as YYYY/MMDDHH:MM:SS.

Note STOPAT, RESTORETOMARK, RESTORETOMARKAFTERTIME, RESTOREBEFOREMARK, and RESTOREBEFOREMARKAFTERTIME are mutually exclusive restore database parameters. If either RESTORETOMARKAFTERTIME or RESTOREBEFOREMARKAFTERTIME are used, then the batch file must also specify a datetime string with the keyword STOPAFTER.

STRIPES	integer	no	1	Number of stripes. Range is 1-32.
STOPAFTER	datetime string	no	none	Specifies datetime for SQL Server 2000 **Restore to mark** options. The datetime string is formatted as YYYY/MMDDHH:MM:SS.
STORAGEIMAGE	string	no	none	Used for restoring a database that was backed up using a grouped Advanced Client snapshot. STORAGEIMAGE identifies the image with which the physical files are associated.
TO	file path	no	none	Specifies a filegroup destination path. Required for each MOVE keyword. Also must sequentially follow each MOVE entry. The value may be delimited with single quotes.
TRACELEVEL	MIN, MID, MAX	no	MIN	Trace level.
TRXOPTION	NOTRUNC, TRUNCONLY	no	none	SQL Server transaction log backup options. If neither option is selected, then the transaction log will be backed up and truncated.
USERID	string	no	sa	Userid for logging into SQL Server.
VDITIMEOUTSECONDS	integer	no	300	Timeout interval for SQL Server Virtual Device Interface

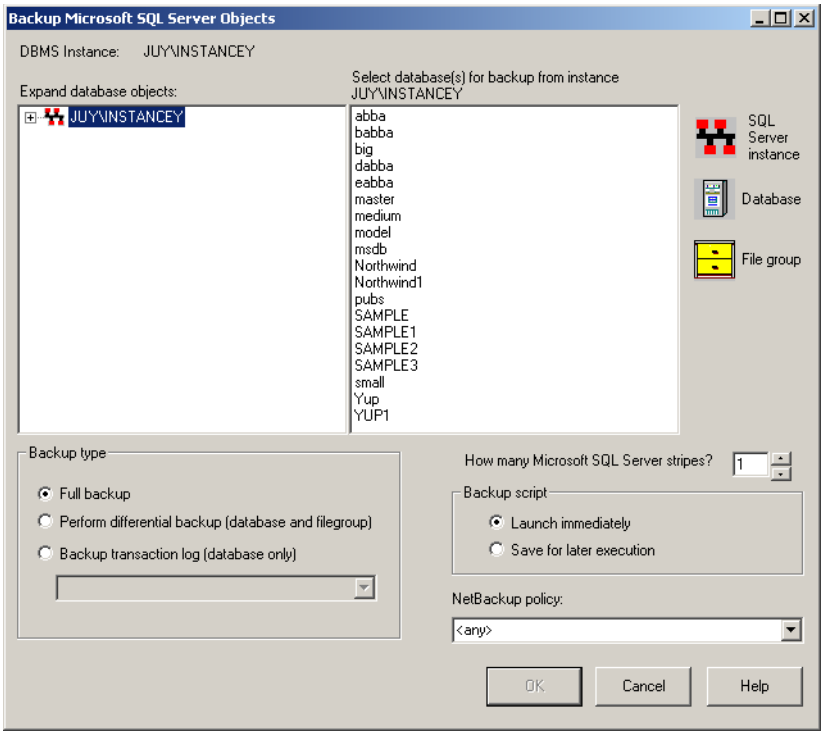
Creating a Backup Batch File

You can use any of the backup or restore dialogs to create a batch file containing a NetBackup for SQL Server script that can be executed at a later time from the Start Batch File dialog, from the `dbbackex` command line program, or by the NetBackup server scheduler.

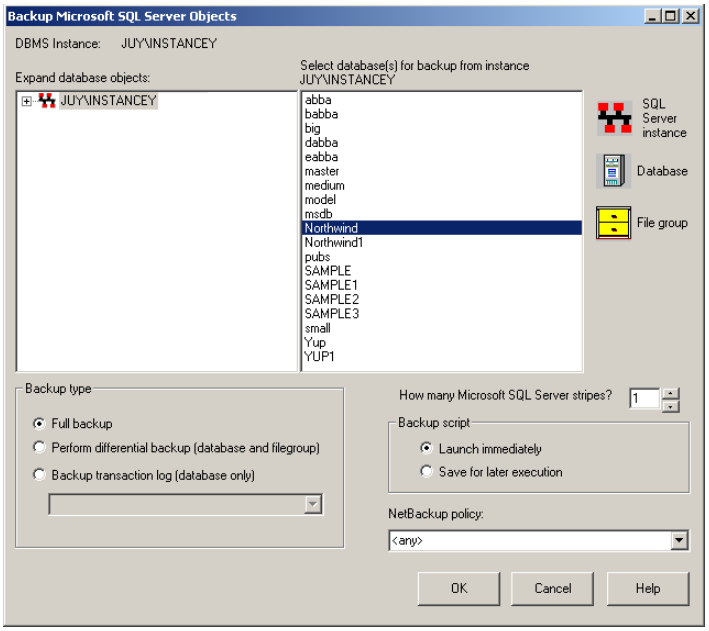
The following example demonstrates how to create and save a script for a simple backup.

1. Choose **Actions** > **Backup**.

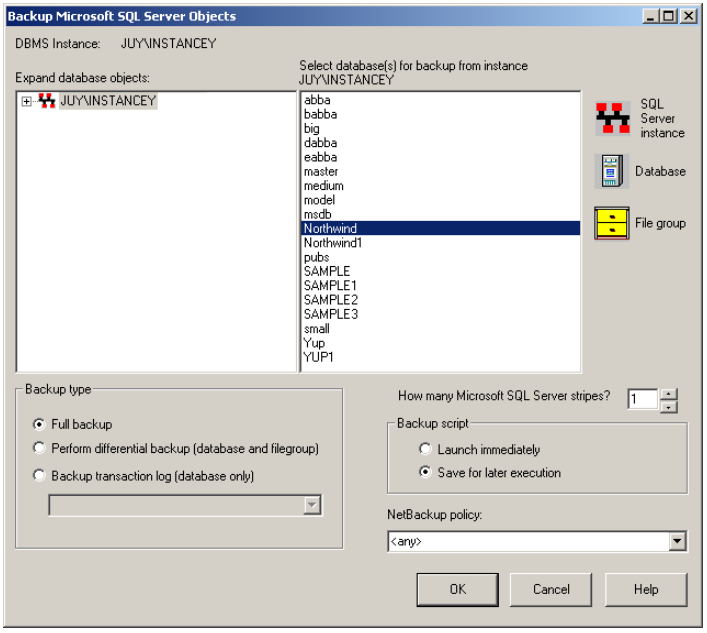
   The Backup Database dialog is displayed.



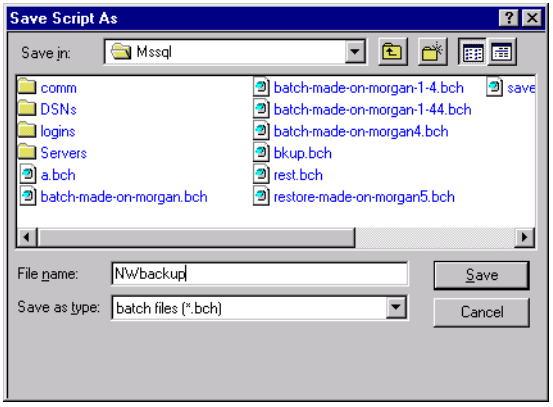
2. Select a database from the **Database** box. For this tutorial, we are going to back up Northwind.



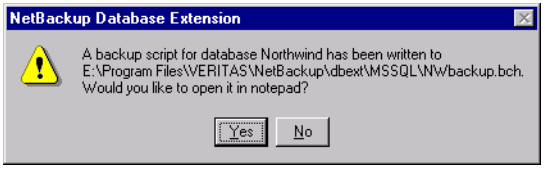
3. In the **Backup Script** radio group, pick **Save for Later Execution**.



4. Click **OK**. Enter a filename and click **Save**.



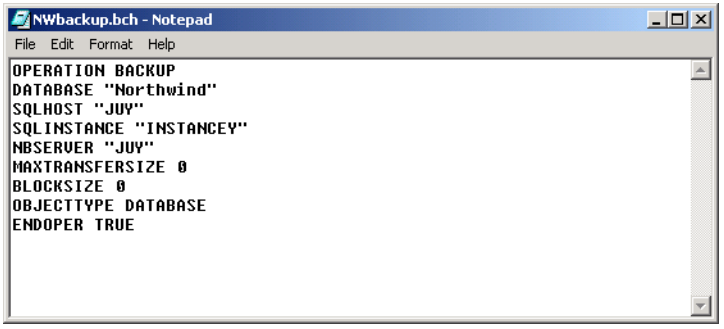
The following message box is displayed.



Note Alternatively, in step 4 you can highlight the name of an existing file, and NetBackup for SQL Server will append the new script to it.

5. Click **Yes**.

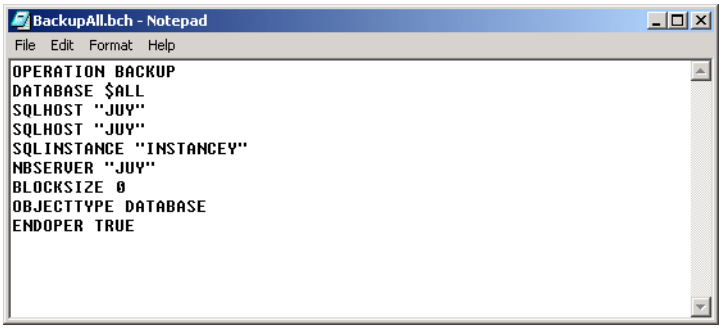
The batch file that you just created will open in Notepad.



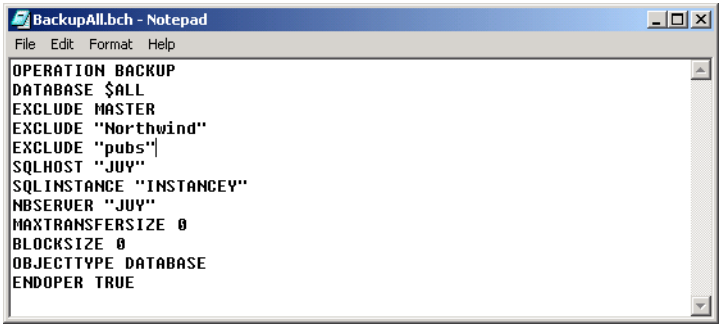
Backing Up All Databases or Excluding Databases

You may wish to back up all of the databases in an instance or to back up all of the databases except for certain ones that you identify. This can be accomplished by generating a backup batch file similar to the example shown in "Creating a Backup Batch File" on page 108, opening it in Notepad, and modifying it.

- To specify that all databases should be backed up, use the DATABASE $ALL keyword/value pair. For example the previous example would be modified as follows.



- To specify all databases except for certain databases that you specify use DATABASE $ALL followed by a series of EXCLUDE <database> keyword/value pairs. For example:



Sample Batch Files

Sample 1 - Simple script to back up a database named BUSINESS.

Note that the parameters for this operation are guided by certain default values. For example, there will be one backup stripe, minimum trace level, and the object type will be a database (as opposed to a transaction log).

```
OPERATION BACKUP
DATABASE BUSINESS
ENDOPER TRUE
```

Sample 2 - Simple script to restore a database named pubs.

This sample restores database *pubs* based upon the backup `NBIMAGE ca.MSSQL7.CA.db.pubs.~.0.001of001.20000203082050..C`. You can find out which backups are available for restore by looking at the `dbclient` log file created when you did the backup or by using `bplist`. Refer to "Using bplist to Retrieve SQL Server Backups" on page 125.

```
OPERATION RESTORE
OBJECTTYPE DATABASE
DATABASE pubs
NBIMAGE ca.MSSQL7.CA.db.pubs.~.0.001of001.20000203082050..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```

Note Use the "Restore Database and Transaction Logs" dialog of the NetBackup Database Extension GUI in order to generate a restore script either for immediate launch or for later execution.

Sample 3 - Perform a striped database backup. Allow multiple internal buffers per stripe.

This example backs up the BUSINESS database using 4 data streams. Each data stream uses 2 buffers.

```
OPERATION BACKUP
DATABASE BUSINESS
STRIPES 4
NUMBUFS 2
SQLHOST "Ca"
NBSERVER "Ca"
ENDOPER TRUE
```

Sample 4 - Restore a database from multiple stripes.

For a striped restore, you must specify the number of stripes and the name of the first backup image name. Notice that the backup image in this example is embedded with the string `.001of004`, which indicates that it is the first of four backups.

```
OPERATION RESTORE
OBJECTTYPE DATABASE
DATABASE Northwind
NBIMAGE ca.MSSQL7.CA.db.Northwind.~.0.001of004.20000216151937..C
STRIPES 004
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```

Sample 5 - Restore a database transaction log up to a point in time.

This script is executed after the database is restored. The database will be restored to the specified point in time (Feb 16, 2000 at 2:03:00 PM), which precedes the date of the backup log (Feb 16, 2000 at 2:03:21 PM).

```
OPERATION RESTORE
OBJECTTYPE TRXLOG
STOPAT 20000216/14:03:00
DATABASE Northwind
NBIMAGE ca.MSSQL7.CA.trx.Northwind.~.0.001of001.20000216140321..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
ENDOPER TRUE
```


Note If STOPAT is not specified, then the database would be restored to the date of the backup log.

Note You can avoid manually staging the restoration of the database backup and the associated log files by using the Restore Microsoft SQL Server Objects dialog of the NetBackup for Microsoft SQL Server GUI.

Note Since RECOVEREDSTATE was not specified, the database will be restored to a recovered state following successful execution of this script.

Sample 6 - Perform an operation and specify the userid and password to use to SQL Server.

Only specify a userid and password if you are using standard SQL Server security. See DBMS (Database Management System) Privileges for more information.

```
OPERATION BACKUP
DATABASE BUSINESS
SQLHOST "Ca"
NBSERVER "Ca"
USERID BOZO
PASSWORD OHMYGOSH
ENDOPER TRUE
```

Sample 7 - Perform multiple operations in sequence.

In this sample batch file, five separate backups will be performed sequentially. Remember that each operation is required to be completely specified.

```
OPERATION BACKUP
DATABASE BUSINESS
OBJECTTYPE DATABASE
SQLHOST "Ca"
NBSERVER "Ca"
STRIPES 5
ENDOPER TRUE

OPERATION BACKUP
DATABASE RECREATION
SQLHOST "Ca"
NBSERVER "Ca"
OBJECTYPE TRXLOG
ENDOPER TRUE

OPERATION BACKUP
```

```
DATABASE EDUCATION
SQLHOST "Ca"
NBSERVER "Ca"
STRIPES 2
ENDOPER TRUE

OPERATION BACKUP
DATABASE GOVERNANCE
SQLHOST "Ca"
NBSERVER "Ca"
OBJECTYPE TRXLOG
ENDOPER TRUE

OPERATION BACKUP
DATABASE SURVIVAL
SQLHOST "Ca"
NBSERVER "Ca"
OBJECTYPE TRXLOG
ENDOPER TRUE
```

Sample 8 - Perform a set of operations in parallel.

This sample is identical to the previous one except that the first operation contains `BATCHSIZE 3`. This tells NetBackup to start the first three operations in parallel, then after these are all completed begin the next set of 3. In this case, since there are just five operations, the second batch set will contain two operations.

```
BATCHSIZE 3
OPERATION BACKUP
DATABASE BUSINESS
SQLHOST "Ca"
NBSERVER "Ca"
OBJECTYPE DATABASE
STRIPES 5
ENDOPER TRUE

OPERATION BACKUP
DATABASE RECREATION
SQLHOST "Ca"
NBSERVER "Ca"
OBJECTYPE TRXLOG
ENDOPER TRUE

OPERATION BACKUP
DATABASE EDUCATION
SQLHOST "Ca"
NBSERVER "Ca"
```


```
STRIPES 2
ENDOPER TRUE

OPERATION BACKUP
DATABASE GOVERNANCE
SQLHOST "Ca"
NBSERVER "Ca"
OBJECTYPE TRXLOG
ENDOPER TRUE

OPERATION BACKUP
DATABASE SURVIVAL
SQLHOST "Ca"
NBSERVER "Ca"
OBJECTYPE TRXLOG
ENDOPER TRUE
```

Sample 9 - Specify the maximum transfer size and block size for a backup.

This sample batch file backs up database business using a maximum transfer size of 64 kilobytes bytes * 2^4 (1M) and a block size of 512 bytes * 2^6 (32 kilobytes).

```
OPERATION BACKUP
DATABASE BUSINESS
SQLHOST "Ca"
NBSERVER "Ca"
MAXTRANSFERSIZE 4
BLOCKSIZE 6
ENDOPER TRUE
```

Sample 10 - Stage a database restore from a database backup, a differential backup, and a series of transaction backups.

This is an example of a script generated by the Restore Microsoft SQL Server Objects dialog.

```
OPERATION RESTORE
OBJECTTYPE DATABASE
DATABASE Northwind
NBIMAGE ca.MSSQL7.CA.db.Northwind.~.0.001of001.20000216135259..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE
```

```
OPERATION RESTORE
OBJECTTYPE DATABASE
DATABASE Northwind
DUMPOPTION INCREMENTAL
NBIMAGE ca.MSSQL7.CA.inc.Northwind.~.0.001of001.20000216135423..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE TRXLOG
STOPAT 20000216/14:03:00
DATABASE Northwind
NBIMAGE ca.MSSQL7.CA.trx.Northwind.~.0.001of001.20000216140149..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE TRXLOG
STOPAT 20000216/14:03:00
DATABASE Northwind
NBIMAGE ca.MSSQL7.CA.trx.Northwind.~.0.001of001.20000216140233..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE TRXLOG
STOPAT 20000216/14:03:00
DATABASE Northwind
NBIMAGE ca.MSSQL7.CA.trx.Northwind.~.0.001of001.20000216140321..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```


Sample 11 - Stage a database restore from a filegroup backup, several file backups, and a series of transaction backups.

This is an example of a full database restore script generated by the Restore Microsoft SQL Server Objects dialog.

```
OPERATION RESTORE
OBJECTTYPE FILE
OBJECTNAME TEST2_Primary
DATABASE TEST2
NBIMAGE ca.MSSQL7.CA.fil.TEST2.TEST2_Primary.0.001of001.20000216143717..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE FILEGROUP
OBJECTNAME TEST2_FG1
DATABASE TEST2
NBIMAGE ca.MSSQL7.CA.fg.TEST2.TEST2_FG1.0.001of001.20000216143846..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE FILE
OBJECTNAME TEST2_FG2_Dat1
DATABASE TEST2
NBIMAGE ca.MSSQL7.CA.fil.TEST2.TEST2_FG2_Dat1.0.001of001.20000216144023..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE FILE
OBJECTNAME TEST2_FG2_Dat2
DATABASE TEST2
NBIMAGE ca.MSSQL7.CA.fil.TEST2.TEST2_FG2_Dat2.0.001of001.20000216144102..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE
```

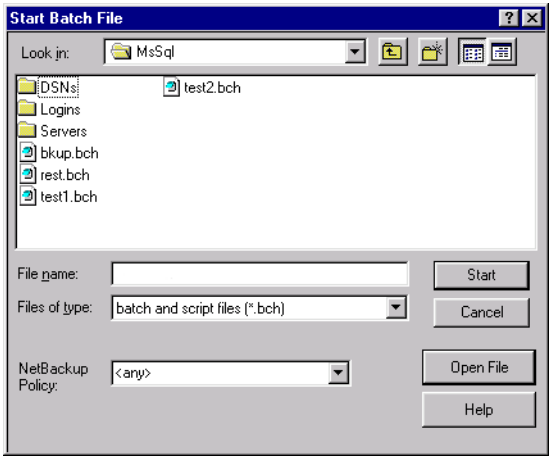
```
OPERATION RESTORE
OBJECTTYPE FILE
OBJECTNAME TEST2_FG2_Dat3
DATABASE TEST2
NBIMAGE ca.MSSQL7.CA.fil.TEST2.TEST2_FG2_Dat3.0.001of001.20000216144126..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE FILE
OBJECTNAME TEST2_FG2_Dat4
DATABASE TEST2
NBIMAGE ca.MSSQL7.CA.fil.TEST2.TEST2_FG2_Dat4.0.001of001.20000216144150..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
STOPAT 20000216/02:44:28
OBJECTTYPE FILE
OBJECTNAME TEST2_FG2_Dat4
DATABASE TEST2
NBIMAGE ca.MSSQL7.CA.trx.TEST2.~.0.001of001.20000216143803..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
STOPAT 20000216/02:44:28
OBJECTTYPE FILE
OBJECTNAME TEST2_FG2_Dat4
DATABASE TEST2
NBIMAGE ca.MSSQL7.CA.trx.TEST2.~.0.001of001.20000216144428..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "Ca"
NBSERVER "Ca"
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```


Running NetBackup Batch Files

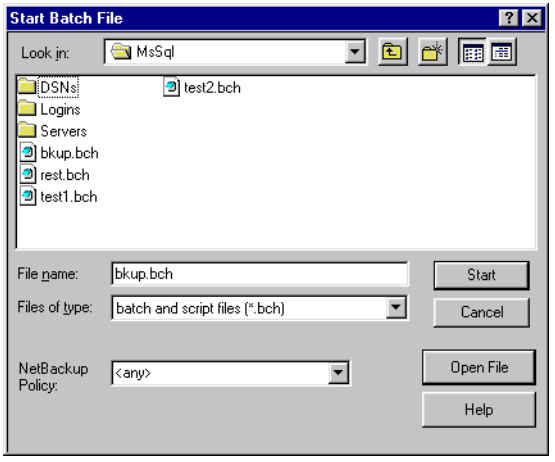
1. Complete the procedure for typing your default login parameters as described in "Starting the NetBackup Database Client Graphical User Interface for the First Time" on page 62.

2. Choose **Actions** > **Batch Files**.

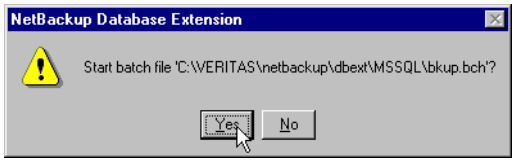
   The following dialog is displayed.



3. Double-click a batch file from the file list. For this tutorial, we are going to select `bkup.bch`. The name of the batch file is displayed in the **File name** box.



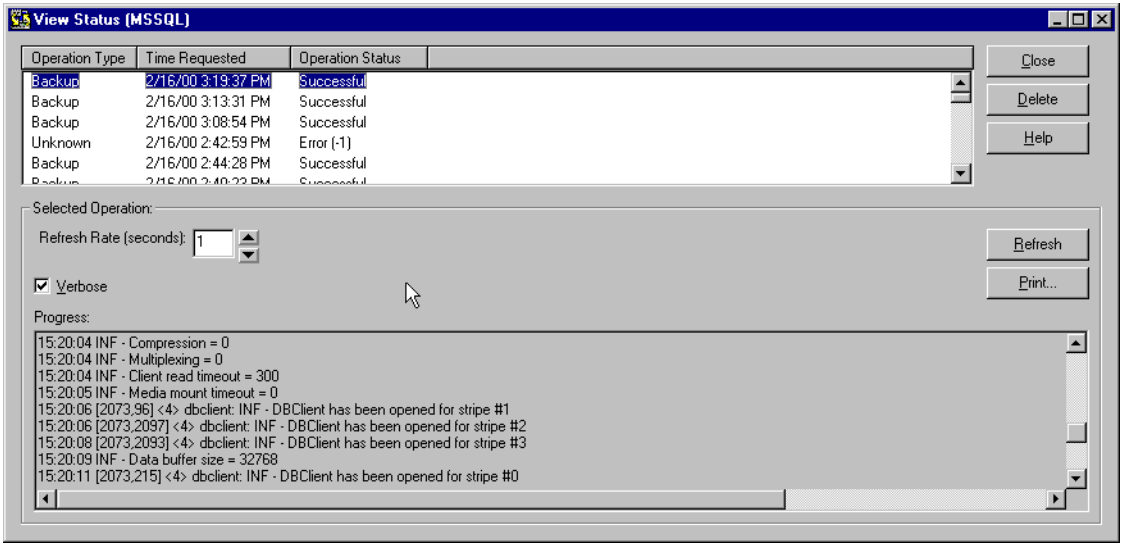
4. Click **Start**. The following is displayed.



5. Click **Yes**.

6. To monitor the operation, choose **View** > **Monitor Jobs**.

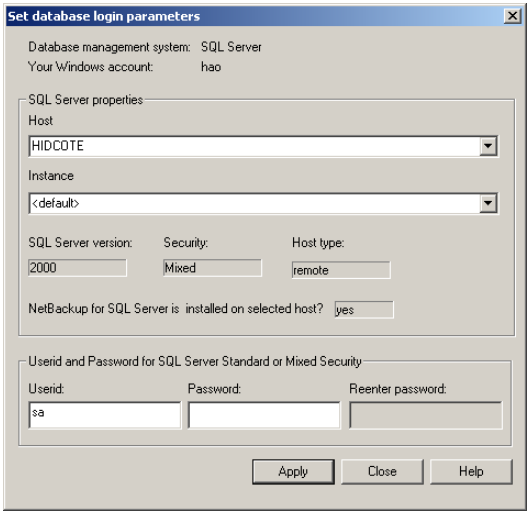
   The following dialog is displayed.



Browsing Objects on Remote SQL Server Installations

You can use NetBackup for SQL Server to browse databases and database backup images on a remote host. Then, you can create batch files for backing up and restoring the SQL Server databases on the remote host. To actually perform the remote backup or restore, you need to save the generated batch file on the remote host and launch the operation from there. You can perform the job launch either by logging onto the remote host; or, you can launch the operation via the NetBackup scheduler.

To browse databases on a remote host, you must select the SQL Server host and instance using the Set database login parameters dialog. Use the drop-down list boxes for **Host** and **Instance** to pick out the ones that you want. After you have selected a remote host/instance, and clicked **Apply**, notice that the **Host type** entry is now "remote."



After selecting a remote database instance from the Set database login parameters dialog, you may also notice that the backup and restore dialogs always open with the Backup (or Restore) Script radio group set to **Save for later execution**, and the **Launch immediately** option disabled. Immediate launch is disabled because the generated script must be executed on the remote host that you are logged on to. After you select **OK** from the backup or restore dialog, use the Save Script As dialog to navigate to the *install_path*\NetBackup\dbext\MSSQL\ folder on the remote host, and save the batch file there.

After saving the batch file on the remote host, you can either launch the batch file from the local installation of NetBackup for SQL Server or you can use the NetBackup Scheduler to launch the batch file. For information about launching a batch file using the NetBackup Database Extension GUI, see "Running NetBackup Batch Files" on page 120. For

information about using the NetBackup Scheduler to launch a batch file, see “Adding New Policies” on page 29, which describes how to create an automatic backup schedule for an MS-SQL-Server backup policy.

Starting Multi-streamed Operations

NetBackup for SQL Server supports backup striping in order to open multiple streams for a single backup or restore. This capability maps directly to functionality supported by the Microsoft version of the SQL backup and restore commands. When you back up an object to multiple streams, NetBackup stores each stream as a separate image. These backups can be multiplexed to a single storage unit or they can each be written to separate storage units depending on how you configured the NetBackup master server. (See "Configuring for Multi-Stream Operations" on page 45 for more information). Either way, all of the backups must be provided simultaneously to SQL Server in order to perform the restore.

Striped backups can be started by setting the "How Many MS SQL Server Stripes" box to a number greater than 1. When you create a batch file yourself, you can invoke backup striping by using the parameter STRIPES and setting it to an integer value greater than or equal to 1 and less than 33.

Restoring backups from multiple stripes is automatic from the Restore Microsoft SQL Server Objects dialog. The backup object will be represented as a single backup with the accompanying notation, "<n> stripes." You simply select the object you wish to restore and NetBackup for SQL Server will find all of the related backups and restore them.

If you wish to create your own batch file to restore a striped object, you need to use the STRIPES keyword to specify the number of stripes that were used on the backup. In addition, for the NBIMAGE keyword, you specify only the name of the first stripe. See "Using bplist to Retrieve SQL Server Backups" on page 125 for more information about backup names used for SQL Server objects.

Using Multiple-Streams for Advanced Backup Methods

If you specify multiple stripes for any Advanced Client backup which streams the frozen image to tape, then NetBackup divides the number of component files equally among the number of stripes. If the number of files is less than the specified number of stripes, then the agent will perform the backup using only as many stripes as there are files.

Note Since Persistent Frozen Image (PFI) policies do not stream the component files to tape, PFI backups ignore the multiple stream directive.

Using bplist to Retrieve SQL Server Backups

This section describes how to use the `bplist` command to obtain NetBackup for SQL Server restore images. This is necessary if you plan to manually create a restore script, rather than through the NetBackup for SQL Server interface.

To extract all of the NetBackup for SQL Server backups from a specific server for a specific client, execute the following command from the MS-DOS prompt.

install_path\NetBackup\bin\bplist -C <*client*> -t 15 -S <*server*> -R \

where <*client*> is the host machine on which NetBackup for SQL Server resides and <*server*> is the host machine of NetBackup server.

See the NetBackup Commands guides for complete information about `bplist`.

The following example shows how to obtain the list of SQL Server backups backed up from client ca to server ca:

Example

```
C:\Program Files\NetBackup\bin\bplist -C juneberry -t 15 -S cole -R \
juneberry.MSSQL7.JUNEBERRY.db.pubs.~.7.001of003.20030920101716..C:\
juneberry.MSSQL7.JUNEBERRY.db.pubs.~.7.002of003.20030920101716..C:\
juneberry.MSSQL7.JUNEBERRY.db.pubs.~.7.003of003.20030920101716..C:\
juneberry.MSSQL7.JUNEBERRY.fil.pubs.pubsnew.7.001of001.20030919175149..C:\
juneberry.MSSQL7.JUNEBERRY\NEWINSTANCE.trx.abc.~.7.001of001.20030902170920..C:\
juneberry.MSSQL7.JUNEBERRY\NEWINSTANCE.fg.abc.PRIMARY.7.001of001.20030902170824.C:\
juneberry.MSSQL7.JUNEBERRY\NEWINSTANCE.db.Howard's Barbeque.~.7.001of001.20030901085255..C:\
juneberry.MSSQL7.JUNEBERRY\NEWINSTANCE.inc.Howard's Barbeque.~.7.001of001.20030903108552..C:\
juneberry.MSSQL7.COLE.db.pubs.~.7.001of001.20030907100101..C:\
juneberry.MSSQL7.COLE.db.pubs.~.7.001of001.20030908200234..C:\
```

Note The colon and backslash, which terminate each line, are not part of the backup name.

The backup name is a string consisting of the following components separated by a delimiter specified by the character preceding the "C" at the end of the backup image name.

1. The host machine where SQL Server resides.
2. The SQL Server identifier
3. The instance name.

   Named instances are formatted as *host**instance-name*. The default instance is simply the name of the host machine.

4. The object type
   - `db`, for database
   - `inc`, for database differential
   - `trx`, for transaction log
   - `fg`, for filegroup
   - `fdg`, for filegroup differential
   - `fil`, for file
5. Database name
6. The name of the file or filegroup if the object type is a file or filegroup; otherwise the symbol ~ is used.
7. The blocksize, specified as 256 kilobytes bytes * 2 ^{blocksize}.
8. <*stripe number*>of<*total stripes*>

   For non-striped backups, this item is always `001of001`. For striped backups, there is one backup for each backup stripe. <*total stripes*> is the total number of stripes for the backup. <*stripe number*> is the count number of the backup for that backup, starting with 001.
9. The timestamp in YYYYMMDDHHMMSS.
10. The character immediately preceding the version indicator is the delimiter, which by default is a period. However, if a period is used in any of the fields, the delimiter may be another character. For example, “B” indicates that the image was created by NetBackup 3.4, prior to 3.4.1. “C” is for NetBackup 3.4.1 and higher.

Disaster Recovery 5

This chapter contains information and instructions on restoring the SQL Server in a disaster recovery scenario.

Preparing for Disaster Recovery of SQL Server

When you are developing your SQL Server disaster recovery plan you will need to plan for how to recover from corruption of the master database as well as loss of your host machine. If the master database has been corrupted, then SQL Server will not start. When this happens you will need to use the SQL Server rebuild master (`rebuildm.exe`) utility to start the SQL Server service. This utility, however, does not recreate the schema information of your application databases. To recover your database schema use the NetBackup SQL Server agent to restore your latest backup of the master database.

As a starting point, disaster recovery of SQL Server assumes that you have already put in place a strategy to recovery from other sorts of data loss such as disk, software, and human error. To prepare for disaster recovery you will need to make frequent backups of the master database, minimally, after you have added or dropped databases or carried out other operations that may result in schema definitions.

Disaster Recovery of SQL Server

SQL Server corrects itself automatically from temporary or minor problems. However, most disasters are beyond the scope of the automatic recovery feature. For example, if a database becomes severely corrupted, or there is a catastrophic failure, recovery must be initiated by the system administrator.

User-initiated recovery could entail either restoring the entire server, including the SQL Server databases, from full system backups, or restoring only the SQL Server databases to a newly-installed or other available SQL Server.

Restoring the entire server has the added benefit of recovering other applications and data which may have resided on the server at the time of failure, and can be accomplished using one of the following methods:

- Manual recovery of the server. This method involves manually restoring the server from full system backups (see “Preparing for Disaster Recovery of SQL Server” on page 128).
- The Intelligent Disaster Recovery option. This option provides a more automated method of restoring the server from full system backups. Refer to the *NetBackup System Administrator’s Guide for Windows, Volume II*, for more information.

Alternatively, the SQL Server databases can be restored to a newly-installed or other available server. This server must be running the same version of Windows on the same hardware platform (cross-platform restores are not supported), the same Service Pack level, and the same version of SQL server with the same service pack as the original server.

For the purposes of disaster recovery, you should only be restoring to a new installation of SQL Server. If you want to restore to an existing installation of SQL Server with other active databases, refer to "Disaster Recovery of SQL Server Databases."

After recovery of the server is complete, or after the new server installation is available, recovery of the SQL Server databases can begin.

Disaster Recovery of SQL Server Databases

If you are restoring to a new SQL Server installation, skip the steps for rebuilding the master database. If you are running SQL Server in a cluster or you are using a version of SQL Server prior to SQL Server 2000, you will need to start SQL Server in single-user mode before restoring the databases.

To recover the SQL Server databases, you need to perform the following tasks:

- rebuild the master database (if restoring to an existing SQL Server)
- start SQL Server in single-user mode (SQL 7.0 or below)
- restore the SQL databases

▼ To rebuild the master database

1. Run the Rebuild Master utility (`\`*`SQL Server installation directory`*`\bin\Rebuildm.exe`).

Note See the Microsoft SQL Server official documentation for information on how to use `rebuildm.exe`.

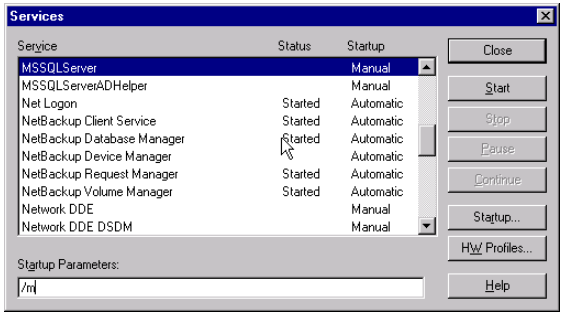
2. When the rebuild is complete, restart the SQL Server services if necessary.

▼ To start SQL Server in single-user mode

Note You need to start SQL Server in single-user mode only if SQL Server is configured in a cluster or if you have a version of SQL Server below SQL Server 2000.

1. From the Windows **Start** menu, choose **Settings** > **Control Panel**.
   - On Windows NT, double-click on **Services**.
   - On Windows 2000 or later, double-click on **Administrative Tools**, then double-click on **Services**.
2. Start the MSSQLServer service or the default instance of SQL Server in single-user mode by typing the following in the **Startup Parameters** box of the Services dialog:

   `/m`



3. Click **Start**.

4. Click **Close** to close the Services dialog.

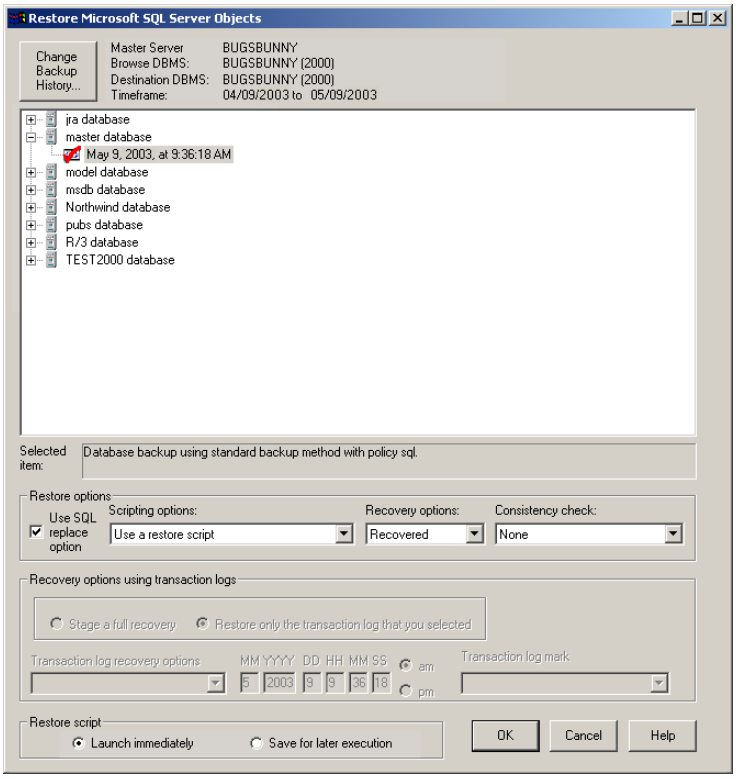
▼ To restore the SQL databases

1. If you have SQL 7.0 or below, you must start SQL Server in single-user mode before restoring the SQL databases. See the previous procedure for details.

2. Open the NetBackup for SQL Client interface.

3. Locate all the media required to perform the restore operations. Choose **Actions** > **Restore**.

4. Select the backup image that contains the copy of the master database to be restored.



Select only the master database at this time. Also, ensure that **Recovered** is checked for the Recovery Options within the Restore Options group.

5. Click **OK**.

6. If you manually placed SQL Server in single-user mode, restart the SQL Server service after the restore has completed.

7. Continue restoring the remaining SQL Server Databases.

   Follow the instructions for restoring SQL databases, differentials, transaction logs, files and filegroups from the "NetBackup Restore Operations" on page 75.

When all of the restore operations have completed successfully, then the recovery of the SQL Server databases is complete.

After the recovery has been completed, it is strongly recommended that a full database backup be performed as soon as possible.

Using NetBackup for SQL Server with Clustering Solutions 6

NetBackup for SQL Server can be used for backing up and restoring installations which are clustered either with Microsoft Cluster Server (MSCS) or VERITAS Cluster Server (VCS).

Support for Microsoft Cluster Server Clusters

NetBackup for SQL Server supports MSCS clusters of 2, 3, or 4 nodes. For SQL 7, there can be either one or two virtual SQL Servers active in the cluster. For SQL 2000 there can be up to 24 active virtual SQL Servers. Unique SQL Server instances in an MSCS cluster are distinguished by the virtual server name. An operation performed with an instance of SQL Server which is clustered with MSCS is very similar to a non-clustered operation except that the client or host name is the SQL virtual name and the designated instance is <*default*>.

If SQL Server shares a cluster with a virtual NetBackup server, then SQL Server and the NetBackup server may be placed in the same group; but they must have different IP and network names.

NetBackup must be installed on each node in the MSCS cluster.

Support for VERITAS Cluster Server Clusters

NetBackup for SQL Server supports VCS clusters with active SQL Server instances. The VirtualName attribute under the VERITAS Cluster Server resource type, Lanman, is the name of the virtual SQL Server. An operation performed with an instance of SQL Server that is clustered with VCS is very similar to a non-clustered operation except that the client or host name is the VirtualName and the designated instance is <default>.

NetBackup must be installed on each node in the VCS cluster.

Installing NetBackup for SQL Server in a Cluster

NetBackup for SQL Server is installed with the server and client software. To use this agent you need to register a valid license key for it on the master or media server. In a clustered environment, the key needs to be registered on each node on which NetBackup Server is installed.

Installation Prerequisites

Before enabling NetBackup for SQL Server, be sure to complete the following procedures. For prerequisites for installing the agent in a cluster, refer to the subsection below, "Installation Prerequisites When Installing in a Cluster."

- Install NetBackup server software on the server.

  Refer to the *NetBackup Installation Guide for Windows* or the *NetBackup Installation Guide for UNIX* for details.

- Install the NetBackup client software on the client where you will be backing up the databases. This step also installs NetBackup for SQL Server.

  See the *NetBackup Installation Guide for Windows* for installation instructions on Windows clients.

Installation Prerequisites When Installing in a Cluster

- The SQL Server must be installed on each node to which NetBackup can failover.
- The NetBackup for Windows server or client must be installed on each node to which NetBackup can failover in the cluster. The NetBackup for SQL Server software is installed along with the server and the client software.

Configuring the NetBackup Server to Be Aware of Clustered SQL Server Instances

Perform the following configuration steps after the installation of a virtual SQL Server (VIRTUALSERVER) has been created and, if applicable, the installation of a virtual NetBackup media server. The following actions must be performed on the master server or on a NetBackup remote client console acting for the master server.

▼ To configure NetBackup Server to be aware of clustered SQL Server instances

1. Using the NetBackup Administration Console, create a MS-SQL-Server policy (for example, VIRTSQLPOLICY), to specify the storage attributes of the backup. Set up the following attributes:

   a. Policy-storage unit: Specify a storage unit belonging to the storage group intended for backup. If you are using a virtual media server, then specify a storage unit belonging to the virtual media server.

   b. Create a backup policy schedule for VIRTSQLPOLICY.

   c. Add the virtual SQL Server name (VIRTUALSERVER) to the client list.

2. To incorporate SQL Server backup scheduling, create an automatic schedule in the VIRTSQLPOLICY. Create the automatic schedule as follows:

   a. Add an automatic backup schedule to VIRTSQLPOLICY.

   b. Add one or more script names (batch files) to the file list.

3. Create a standard backup policy (say, STDPOLICY). Add all physical names in the cluster to the client list of STDPOLICY.

4. Establish the permissions settings as follows for a redirected restore to a different client: On the master server, either:

   - Create a file called *install_path*\NetBackup\db\altnames\No.Restrictions

     OR

   - Create each of the files, *install_path*\NetBackup\db\altnames\NODEA, *install_path*\NetBackup\db\altnames\NODEB.

Note Creating the `No.Restrictions` file allows all clients to perform redirected restores to different clients. This parameter may need to be added and removed according to site policies.

Performing a Backup on a Virtual SQL Server Instance

Although NetBackup for SQL Server supports browsing for databases and images on any of the physical nodes, a user backup or restore will not be successful unless it is launched from the NetBackup client that is active. After NetBackup has been configured as described in the previous sections, backing up a SQL Server database from a virtual instance is similar to backing up a database from a non-virtual one.

▼ To perform a backup on a virtual SQL Server instance

1. Open the NetBackup for SQL Server interface on the active platform node.
2. Choose **Actions** > **Backup**.
3. Select one or more databases.
4. In the **NetBackup Policy** field, enter the name of the MS-SQL Server policy (VIRTSQLPOLICY) that was created for specifying storage attributes for the virtual SQL Server backup.
5. Click **OK**.

Performing a Restore on a Virtual SQL Server Instance

▼ To perform a restore on a virtual SQL Server instance

1. Open the NetBackup for SQL Server interface on the active platform node.
2. Choose **Actions** > **Restore**.

   The Backup History Options dialog is displayed.
3. From the **SQL Host** list, select the Virtual Server name (VIRTUALSERVER) as the SQL host.
4. From the **SQL Server Instance** drop-down list, choose **<default>**.
5. Click **OK**.

   The Restore Microsoft SQL Server Objects dialog is displayed.
6. Select a backup image or staged image list.
7. Click **OK**.

Backing Up SQL Server in a Cluster with a Multi-Interface Network Connection

This section covers information that is required for combining SQL Server clustering with the usage of public-private interfaces to perform backups. Many administrators desire to reserve a separate network interface for their SQL Server host machines that is used for routing backup traffic. This type of configuration requires special consideration in terms of configuring both the NetBackup master server and the NetBackup client which backs up SQL Server. In addition, a special consideration is required in terms of how to browse for SQL Server backup images.

The following distinct network resources can be identified in a dual-interface SQL Server cluster.

- the public name of each SQL Server host (for example, SQLHOST1 and SQLHOST2),
- the private interface name used for backing up each of the SQL Server hosts (for example, SQLHOST1-NB and SQLHOST2-NB),
- the public virtual name of the SQL Server (for example, VIRTSQL), and
- the private virtual name of the SQL Server (for example, VIRTSQL-NB).

Master Server Configuration

Two configuration changes must be made on the master server to allow for backups and restores over a private interface: the backup policies must include the private interface name in the Clients list of the policy and permissions must be added to allow for browsing of backups across the private interface.

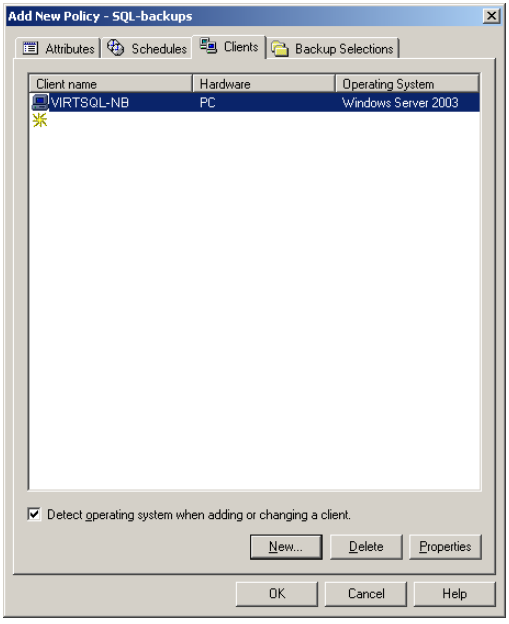
Adding Clients to the Policy

The private name of the client must be added to the Clients list of the policy. The NetBIOS or public name of the client should not be used.

▼ **To add clients to the policy**

1. Open the NetBackup Administration Console.
2. Create a new policy or open an existing policy.

In the Clients list for the policy, add a new client. Specify the Client name as the private SQL virtual instance name, for example, VIRTSQL-NB.



Adding Permissions That Allow for Browsing of Backups Across the Private Interface

The administrator can allow all clients or allow single clients to browse and restore backups performed over the multi-nic connection.

▼ To allow all clients to browse for backups and perform restores

- Add the empty file `NB_INSTALL\db\altnames\No.Restriction` on the master server.

  This option allows any NetBackup client to browse backups made by the SQL Server host machine.

▼ To allow a single client to browse for backups and perform restores

- Add the empty file, `NB_INSTALL\db\altnames\`*`SQLHOST`* on the master server.

  This only allows the SQL Server host machine, SQLHOST, to access backups managed by the master server.

Client Configuration

The private interface name of the SQL Server host machine must be specified in the Backup, Archive, and Restore interface on the client. The host file must also be changed for each node in the cluster.

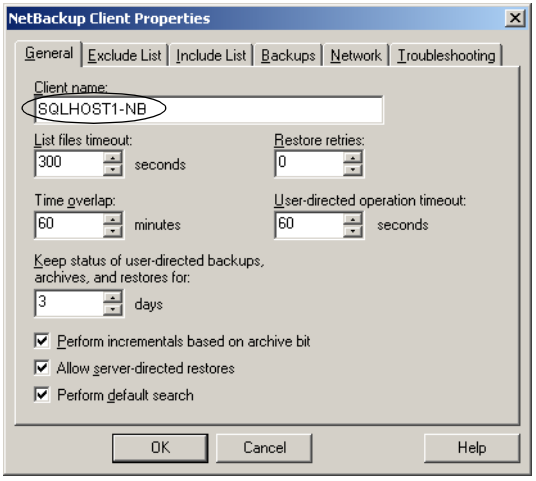
Specifying the Private Name of the Client

The Client name can be set either during the installation of the NetBackup client, or it can be changed in the NetBackup Client Properties dialog in the Backup, Archive, and Restore interface, as described below.

▼ To specify the private name of the client

1. Open the Backup, Archive, and Restore interface.
2. Choose **File** > **NetBackup Client Properties**.
3. Click on the **General** tab.
4. In the **Client name** box, specify the private name of the client.

   For example, the client name used for the SQLHOST1 machine would be SQLHOST1-NB and the client name used for the SQLHOST2 machine would be SQLHOST2-NB.



Modifying the Host File on Each SQL Server Host

The host file must be changed on each node in the cluster. Perform the following procedure for each SQL Server host machine in the configuration.

▼ **To modify the host file on each SQL Server host**

- ❖ Modify the local host file that specifies the private network interfaces. For example:

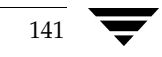
```
192.168.0.1   SQLHOST1-NB   Private name of node 1
192.168.0.2   SQLHOST2-NB   Private name of node 2
192.168.0.3   VIRTSQL-NB   Private name of SQL instance
```

Performing Backups in a Cluster with a Multi-Nic Connection

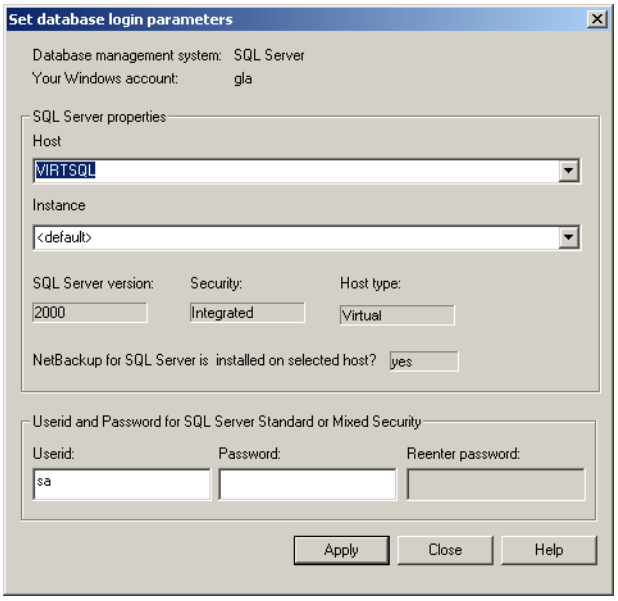
To perform backups of SQL Server using a multi-nic connection, you need to specify the public name of the *virtual* SQL Server host in the Set database login parameters dialog. You also need to create and edit a batch file for backups so that it includes the private virtual SQL Server name.

▼ **To perform backups**

1. On either SQLHOST1 or SQLHOST2, open the NetBackup for SQL Server interface.
2. Choose **Options** > **Set DBMS login parameters**.



3. In the **Host** box, specify the public name of the virtual SQL Server host (VIRTSQL).



4. Choose **Actions** > **Backup**.

5. Select the databases to back up.

Note Do not attempt to perform an immediate backup from the backup dialog. The generated batch files must be modified before they can be run successfully.

6. Select the desired backup options and choose **Save for later execution**.

7. Click **OK**.

   A batch file similar to the following will be created:

```
OPERATION BACKUP
DATABASE "ACCOUNTING"
SQLHOST "VIRTSQL"
NBSERVER "THOR"
BROWSECLIENT "VIRTSQL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
ENDOPER TRUE
```

8. Change the line value associated with BROWSECLIENT from the public virtual SQL Server name to the private name.

```
OPERATION BACKUP
DATABASE "ACCOUNTING"
SQLHOST "VIRTSQL"
NBSERVER "THOR"
BROWSECLIENT "VIRTSQL-NB"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
ENDOPER TRUE
```

9. Place the modified batch file on both nodes in the cluster so that it will be available for scheduled backups, regardless of which node is active when a backup is initiated.

Performing Restores in a Cluster with a Multi-Nic Connection

To perform restores of SQL Server in a multi-NIC, cluster environment, you need to specify the public name of the virtual SQL Server host in the Set database login parameters dialog and both the virtual SQL Server host name and the private interface name of the virtual SQL Server host in the Browse History Options dialog. You also need to create a batch file for restores and manually edit it to include the private name of the virtual SQL Server.

▼ To perform restores

1. On either SQLHOST1 or SQLHOST2, open the NetBackup for SQL Server interface.
2. Choose **Options** > **Set DBMS login parameters**.
3. In the **Host** box, specify the public name of the virtual SQL Server host (VIRTSQL).
4. Choose **Actions** > **Restore**.
5. In the Backup History Options dialog, do the following.
   a. In the **SQL Host** box, specify the public name of the virtual SQL Server (VIRTSQL).
   b. In the **Source Client** box, specify the private name of the virtual SQL Server (VIRTSQL-NB).
6. Click **OK**.

7. Select the databases to restore.

Note Do not attempt to perform an immediate restore from the restore dialog. The generated batch files must be modified before they can be run successfully.

8. Select the desired restore options and choose **Save for later execution**.

9. Click **OK**.

   A batch file similar to the following is generated.

```
OPERATION RESTORE
OBJECTTYPE DATABASE
DATABASE "ACCOUNTING"
NBIMAGE
"SQLHOST1.MSSQL7.VIRTSQL.db.ACCOUNTING.~.7.001of001.20040306111309..C"
SQLHOST "VIRTSQL"
NBSERVER "THOR"
BROWSECLIENT "VIRTSQL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```

10. Change the line value associated with BROWSECLIENT from the public virtual SQL Server name to the private name.

```
OPERATION RESTORE
OBJECTTYPE DATABASE
DATABASE "ACCOUNTING"
NBIMAGE
"SQLHOST1.MSSQL7.VIRTSQL.db.ACCOUNTING.~.7.001of001.20040306111309..C"
SQLHOST "VIRTSQL"
NBSERVER "THOR"
BROWSECLIENT "VIRTSQL-NB"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```

11. Run the modified batch file by choosing **Actions** > **Batch Files**.

Using NetBackup for SQL Server with a Multi-Interface Network Connection 7

Many administrators desire to reserve a separate network interface for their SQL Server host machines that is used for routing backup traffic. This type of configuration requires special consideration in terms of configuring both the NetBackup master server and the NetBackup client which backs up SQL Server. In addition, a special consideration is required in terms of how to browse for SQL Server backup images.

In the following examples, the NetBackup client machine that backs up SQL Server is referred to as SQLHOST and the private interface used for backing up SQLHOST is referred to as SQLHOST-NB.

Refer to the chapter "Backing Up SQL Server in a Cluster with a Multi-Interface Network Connection" on page 138 provides a description of how to protect SQL Server in a cluster.

Master Server Configuration

Two configuration changes must be made on the master server to allow for backups and restores over a private interface: the backup policies must include the private interface name in the Clients list of the policy and permissions must be added to allow for browsing of backups across the private interface.

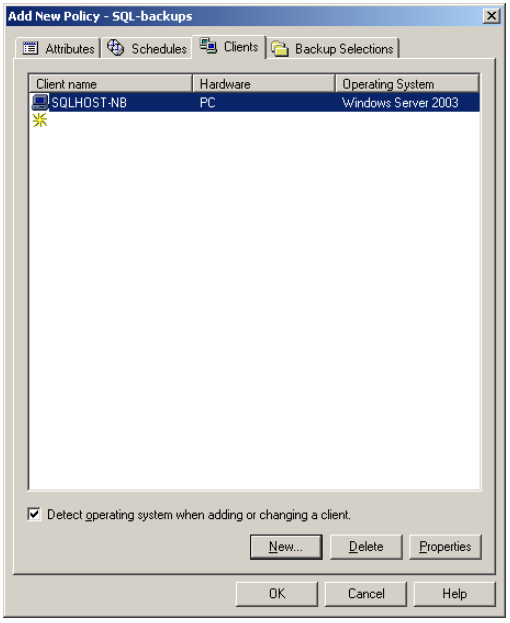
Adding Clients to the Policy

The private name of the client must be added to the Clients list of the policy. The NetBIOS or public name of the client should not be used.

▼ To add clients to the policy

1. Open the NetBackup Administration Console.
2. Create a new policy or open an existing policy.

   In the Clients list for the policy, add a new client. Specify the Client name as the private interface name (in this example, SQLHOST-NB).



Adding Permissions That Allow for Browsing of Backups Across the Private Interface

The administrator can allow all clients or allow single clients to browse and restore backups performed over the multi-nic connection.

▼ **To allow all clients to browse for backups and perform restores**

- Add the empty file `NB_INSTALL\db\altnames\No.Restriction` on the master server.

  This option allows any NetBackup client to browse backups made by the SQL Server host machine.

▼ **To allow a single client to browse for backups and perform restores**

- Add the empty file, `NB_INSTALL\db\altnames\`*`SQLHOST`* on the master server.

  This only allows the SQL Server host machine, SQLHOST, to access backups managed by the master server.

Client Configuration

The private interface name of the SQL Server host machine must be specified in the Backup, Archive, and Restore interface on the client.

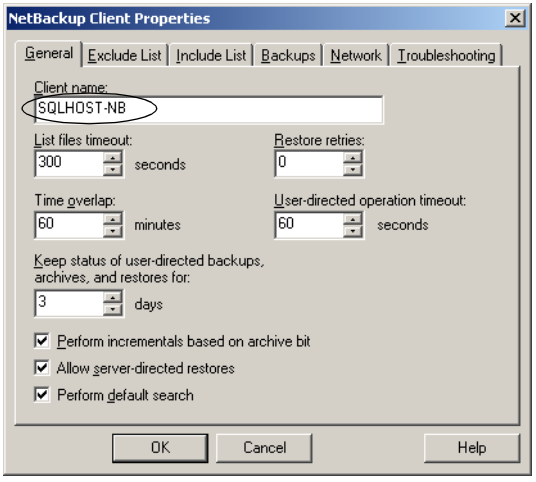
Specifying the Private Name of the Client

The Client name can be set either during the installation of the NetBackup client, or it can be changed in the NetBackup Client Properties dialog in the Backup, Archive, and Restore interface, as described below.

▼ **To specify the private name of the client**

1. Open the Backup, Archive, and Restore interface.
2. Choose **File** > **NetBackup Client Properties**.
3. Click on the **General** tab.
4. In the **Client name** box, specify the private name of the client.

For example, the private name for the machine SQLHOST would be SQLHOST-NB.

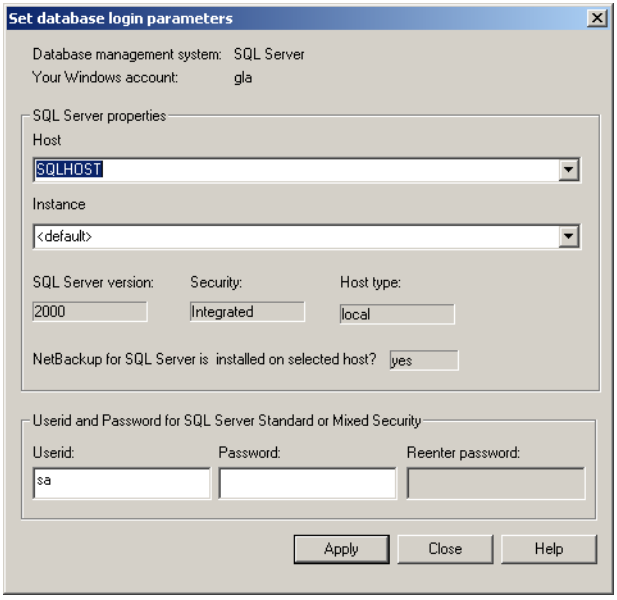


Performing Backups with a Multi-Nic Connection

To perform backups of SQL Server using a multi-nic connection, you need to specify the public name of the SQL Server host in the Set database login parameters dialog.

▼ To perform backups

1. Open the NetBackup for SQL Server interface.
2. Choose **Options** > **Set DBMS login parameters**.
3. In the **Host** box, specify the public name of the SQL Server host (SQLHOST).



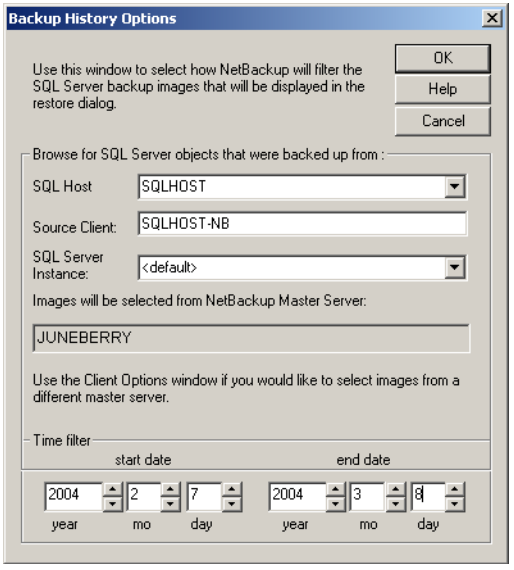
4. Choose **Actions** > **Backup**.
5. Continue with the backup as you would normally.

Performing Restores with a Multi-Nic Connection

To perform restores in multi-NIC environment, you need to specify the public name of the SQL Server host in the Set database login parameters dialog. You also need to provide both the SQL Server host name and the private interface name in the Browse History Options dialog.

▼ To perform restores

1. Open the NetBackup for SQL Server interface.
2. Choose **Options** > **Set DBMS login parameters**.
3. In the **Host** box, specify the public name of the SQL Server host (SQLHOST).
4. Click **OK**.
5. Choose **Actions** > **Restore**.
6. In the **SQL Host** box, specify the primary (or NetBIOS) name.
7. In the **Source Client** box, specify the private interface name.



8. Click **OK**.

   A dialog opens that shows the SQL Server backups of the SQL Host made on the private network interface.

9. Continue with the restore as you would normally.

Performing Backups and Restores in a SAP Environment 8

With NetBackup you can perform scheduled SAP backups, in accordance with a predefined backup strategy, or manual backups that are not planned and may be necessary in exceptional situations. The practices described here are based on the practices recommended by SAP in SAP/MS SQL Server 2000 DBA in CCMS.

The NetBackup backup and restore procedures for the SAP R/3 database are identical to the NetBackup procedures with any other SQL Server database.

NetBackup for SQL Server allows you to create scripts in order to perform full or differential backups of databases and backups of transaction logs. In addition to the database backups and restores, NetBackup also provides the capabilities to backup the SAP file systems.

Manual Backups

The administrator on the master server can use the NetBackup Administration Console to manually execute an automatic backup schedule for an "MS-SQL-Server" policy, where the R/3 database is specified in the backup script. For more information, see the section on manual backups in the *Netbackup System Administrator Guide for Windows*.

Policy Configuration

In order to automatically perform backups of a SAP environment, you need to create backup policies. A backup policy with the "MS-SQL-Server" policy type selected must be created for R/3 database backups. Batch files, which will initiate the backup of the database and transaction logs, must be added to the Selections list in the policy. For information on creating the batch files needed, see "Creating Batch Files to be Used in Automatic Backup Schedules" below. For information on MS-SQL-Server policies, see the instructions in "Adding New Policies" on page 29.

For backups of the executables disk (a file-system backup), a backup policy must be created with the "MS-Windows-NT" policy type selected. For information on MS-Windows-NT policies, see the *Netbackup System Administrator Guide for Windows, Volume 1*.

Creating Batch Files to be Used in Automatic Backup Schedules

NetBackup for SQL Server uses batch files to initiate database backup and restore operations. A batch file must be created for database backups and for transaction log backups. These batch files must then be added to the Selections list in the backup policies you've created.

▼ To create a script for database backups

1. From the Windows **Start** menu, select **Programs**, **NetBackup**, then **NetBackup MS SQL Client**.

   The NetBackup for Microsoft SQL Server Graphical User Interface will open.

2. Choose **Actions** > **Backup**.

   The Backup Microsoft SQL Server Objects dialog box will appear.

   Select **Perform differential backup** if you want to perform a differential backup, rather than a full backup.

3. Select the R/3 database in the **Database** list.

4. For the **Backup type**, choose **Full backup.**

5. Select **Save for later execution** to save the generated script for scheduled backups.

   Select **Launch immediately** in the **Backup Script** group to run the backup immediately.

6. Click **OK**.

7. Specify a filename and click **Save** if the **Save for later execution** option was set.

8. If the **Launch immediately** option is set, then choose **View** > **Monitor Jobs**.

Note You must check that each backup operation is completed successfully. See "Monitoring Backups" on page 156 for more information.

▼ To create a script for transaction log backups

1. Before starting a transaction log backup, the database administrator should set the **Transaction Log Backup Options** database option to off. This is an option on the SQL Server interface that applies to the databases.

   The entire sequence of transaction logs generated following any database dump must be maintained on the same NetBackup server. NetBackup for SQL Server requires that you follow these guidelines in devising your backup strategy in order to ensure success in restoring your database.

2. Choose **Actions** > **Backup**.

   The Backup Microsoft SQL Server Objects dialog appears.

3. Select the R/3 in the **Database** drop-down list.

4. For the **Backup type**, select **Backup transaction log**.

5. Select **Save for later execution** to save the generated script for scheduled backups.

   Select **Launch immediately** in the **Backup Script** group to run the backup immediately.

6. Click **OK**.

7. Specify a filename and click **Save** if the **Save for later execution** option was set.

8. If the **Launch immediately** option is set, then choose **View** > **Monitor Jobs**.

Note You must check that each backup operation is completed successfully. See "Monitoring Backups" on page 156 for more information.

Monitoring Backups

Scheduled backups should be checked regularly to ensure they were completed successfully.

Always check that:

- The most recent backup has run successfully. See "Progress Reports Created for NetBackup for SQL Server on the Client" on page 202.
- All the backups in the backup cycle are being executed according to the schedule. Gaps in a backup sequence can have serious consequences in a subsequent attempt to restore the database.

Restoring the R/3 Database

This section describes how to restore the R/3 database.

- If you have scheduled *differential* backups, review the information in next section "Including Differential Backups in a Restore Operation"Including Differential Backups in a Restore Operation.
- If the R/3 database disk system is damaged or the transaction log disk system is damaged, follow the instructions in "Restoring the R/3 Database after a Disk Crash" on page 157Restoring the R/3 Database after a Disk Crash.
- To perform a regular restore of the R/3 database, follow the instructions in "Restoring the Database and Transaction Log Backups," Restoring the Database and Transaction Log Backups.

Including Differential Backups in a Restore Operation

If you incorporated *differential backups* in the backup strategy, the restore process differs depending on the type of backups available.

- If differential backups were made after the last full database backup, restore the last database backup followed by the most recent differential backup. Then apply all subsequent transaction logs.
- If no differential backups were made since the last full database backup, restore the last full database backup and then apply all subsequent transaction logs.
- If several differential backups are available, but the latest one cannot be read.

  Restore the most recent full database backup and then you restore the latest readable differential backup. Then apply all subsequently created transaction logs.

Restoring the R/3 Database after a Disk Crash

This section describes how to restore the database when the R/3 database disk system is damaged or the transaction log disk system is damaged. This process is only applicable to a configuration with three disk systems: one system for the R/3 database, one for the R/3 transaction logs and one for all others.

Note The R3 database must not be in use when performing a restore operation. Make sure that all SAP services are stopped before attempting a restore using NetBackup.

▼ To restoring the R/3 database after a disk crash

1. Back up the current transaction log.

   If the disk system on which the R/3 database resides is damaged, it is vital to immediately backup the currently active transaction log to prevent loss of data. Without a backup of the current log, the database can only be restored to the status at the time of the last transaction log backup. If work has been carried out on the R/3 system since then, this work will be lost.

2. Replace damaged disks.

   Replacing damaged disks in a RAID disk system is normally a straightforward procedure. If you are uncertain how to proceed, refer to the documentation of your hardware vendor to find out how to handle the disks. The new disks must be formatted and assigned the same drive letter as the old disks.

3. Restore the database and transaction logs.

   The central phase of a restore operation is the reloading of the database backup and the application of the available transaction logs. When the database backup is reloaded, the database files are automatically recreated and the data is copied from the backup device to the newly created files. Once this has been done, the transaction logs are applied in the same sequence as they were originally made. In a final step, open transactions that were not completed at the time of the database failure are rolled back.

Restoring the Database and Transaction Log Backups

NetBackup MS-SQL Server Agent GUI provides for automatic staging. By selecting the latest transaction log backup, the GUI automatically restores the previous full database backup along with any optional differential backups and subsequent transaction log backups. There is also the option to specify a point in time to which to restore to.

Note The R3 database must not be in use when performing a restore operation. Make sure that all SAP services are stopped before attempting a restore using NetBackup.

Caution To restore the R/3 database you first restore the most recent database backup and then the subsequent transaction logs. During the entire procedure, do not execute any transactions and do not shut down the database server. A server shutdown would write a checkpoint to the log and, as a result, you would not be able to restore further transaction logs.

▼ To restore the database and transaction log backups

1. Restore the most recent database backup.
2. Restore the latest differential database backup (if available).
3. Restore all succeeding transaction log backups.
4. Restore the latest transaction log backup.

Using NetBackup for SQL Server with Advanced Client 9

This chapter explains how to use the NetBackup Advanced Client feature in conjunction with the NetBackup SQL Agent to employ snapshot technology for backing up and restoring SQL Server objects. To use NetBackup Advanced Client with NetBackup for SQL Server, you must register valid license keys for both Advanced Client and Microsoft SQL Server.

This chapter contains the following information:

- "NetBackup Advanced Client for SQL Server Overview"
- "How SQL Server Operations Use Advanced Client"
- "Configuring a Snapshot Backup"

NetBackup Advanced Client for SQL Server Overview

Advanced Client, when used in conjunction with NetBackup for SQL Server, enables you to back up and restore SQL Server objects by taking snapshots of the component files and then backing up the snapshot version to tape or committing them to another form of persistent storage. The snapshot technology uses SQL Server VDI (virtual device interface) quiescence to affect a momentary freeze on database activity while NetBackup takes a snapshot of the identified files.

The term database freeze is interchangeable with database quiescence and refers to the momentary state in which all activity in the database has been halted while a snapshot (or frozen image) copy is created. The state-change in which the freeze is lifted is referred to as database thaw.

The role of NetBackup in managing backup images taken from snapshots is common for both Advanced Client and standard database backups. In so far as NetBackup provides an identical view for cataloging and browsing images and for accessing them in restore operations, it is transparent whether the images may have been created by standard database backups or by the Advanced Client. This allows for a similar operational experience whether the user is backing up and restoring standard SQL Server database images or images taken from snapshots. However, there are fundamental differences in the technologies.

Advanced Client backups, in contrast to standard backups, are file-based. This means that NetBackup determines the file list constituting the SQL Server object and backs it up asynchronously with respect to SQL Server. On the other hand, standard backups are stream-based, which means that SQL Server provides data to NetBackup buffer-by-buffer constituting a backup stream. The key role of SQL Server in file-based backups is to provide the mechanism for freezing database activity long enough for NetBackup to invoke a so-called snapshot provider that creates volume snapshots of the files.

The snapshot method is the software/hardware component that creates the snapshot image of the files. The method may be a VERITAS product such as Volume Manager, an operating system component such as VSS (Microsoft Virtual Shadow Service), or a third-party product. Snapshot methods use a variety of technologies such as copy-on-write, split-mirror, or RAID. They also provide several competing methodologies that may have different consequences in respect to factors such as recovery speed, impact of the backup on the client processor, and cost of specialized disks or processing units.

Advanced Client Features

The following NetBackup Advanced Client features are available for use with NetBackup for SQL Server:

Snapshot Backup	A snapshot is a disk image of the client's data made almost instantaneously. NetBackup backs up the data from the snapshot image, not directly from the client's primary data.
Instant Recovery (formerly Persistent Frozen Image or PFI)	An instant recovery backup is similar to a snapshot backup except that the snapshot is not transferred to tape. The restore may be made directly from snapshot copy.
Persistent Frozen Image with Backup to Tape	This feature is similar to the *Instant Recovery* feature except that a backup copy is made to a tape storage unit. If the image is needed for a restore, then a disk copy will be used unless it has been swapped out due to storage contention. In this case, the restore will fall back to the tape copy of the backup.
Alternate Client Backup	An alternate client backup shifts the burden of backup processing to an alternate client, reducing the impact on the client's computing resources ordinarily caused by a local backup. The backup agent reads the data from the shared disk and writes it to storage.

Although all of these features are provided through Advanced Client support for SQL Server, not all methods are supported.For information on selecting a method, see the *NetBackup Advanced Client System Administrator's Guide*. For a description of snapshot methods available for use with NetBackup for SQL Server, see the supported platforms matrix on the VERITAS support site.

Supported Platforms

▼ To access the supported platforms information

1. Go to the VERITAS support web page: `www.support.veritas.com`
2. In the left margin of the VERITAS Support page, choose **Compatibility Lists**. The Online Compatibility Lists page appears.
3. From the Product List, choose **+NetBackup Products**. The list refreshes showing a list of NetBackup products.
4. Choose a NetBackup product. This page contains various links to supported platform lists.

How SQL Server Operations Use Advanced Client

This section includes information on how NetBackup for SQL Server works with the Advanced Client option, including:

- how a backup method is chosen
- what is backed up in a snapshot backup
- what affects the performance of snapshot backups of SQL Server
- how to perform SQL Server snapshot backups
- how to perform SQL Server snapshot restores
- how multiple databases can be backed up in a single snapshot
- how to use a cloaked database snapshot to impact the basing of differential backups

Selection of Backup Method

The selection of a backup methodology, whether standard or Advanced Client, is dependent on what policy is used. If a policy configured for Advanced Client is selected, then additional attributes relating to the policy determine the Advanced Client features and specific snapshot methods used. You can specify a policy name in the Backup Microsoft SQL Server Objects dialog (choose **Actions** > **Backup**). Or specify <any>, if you want NetBackup to choose the policy.

Due to SQL Server limitations certain objects cannot be backed up via snapshots. These are database differentials, filegroup differentials, and transaction logs. If an Advanced Client policy is selected to back up one of these object types, then NetBackup will perform a stream-based backup using the storage unit provided in the policy configuration. If a storage unit is not provided, then NetBackup will use the default storage unit for the server.

What is Backed Up

Although the database administrator works exclusively with logical objects, such as databases and filegroups, it is useful to gain some appreciation of the differences between file- and stream-based backups in terms of the data content that is archived. For stream-based backups, NetBackup simply captures the data stream content provided by SQL Server. If the user has specified multiple streams, then SQL Server opens multiple streams that NetBackup catalogs as separate images.

For file-based backups, NetBackup creates a file list consisting of all the physical files that constitute the object and supplies it to the Advanced Client, which is responsible for snapshot creation. If multiple streams have been specified, then NetBackup divides the

file list into sub-lists. Each sub-list is backed up separately and constitutes a separate image. Users may notice that if multiple streams are specified for a file-based backup and if the number of streams exceeds the number of component files, then the number of file-based streams will not exceed the number of files. With stream-based SQL Server backups, SQL Server always creates exactly the number of streams that the end user specifies.

The file list used for backing up a SQL Server database consists of the physical files constituting the primary filegroup, any secondary filegroups, and the transaction log. Typically, these can be identified respectively by their name extensions, which are `.mdf`, `.ndf`, and `.ldf`. The file list for a filegroup backup consists of the physical files belonging to the filegroup. And, finally, the file list for a SQL Server file object backup consists of a single physical file, namely, the file that maps to the SQL Server file object.

Performance Considerations

When a physical file is backed up using the Advanced Client, the backup will consist of the entire extent. This contrasts with stream-based SQL Server backups where only the actual data content of the objects are archived. For this reason, if you intend to use snapshot technology for backing up SQL Server, then it may be advantageous to use the SQL Server dynamic file allocation to reduce the likelihood that any of the component files contain large areas of empty space.

Another consideration for choosing between file- and stream-based backups concerns how SQL Server zeroes the target disk area prior to a stream-based restores. In some cases, this almost equals the total disk-copy time for restore. For Advanced Client restores, however, disk zeroing is not done so the total recovery time can be substantially less.

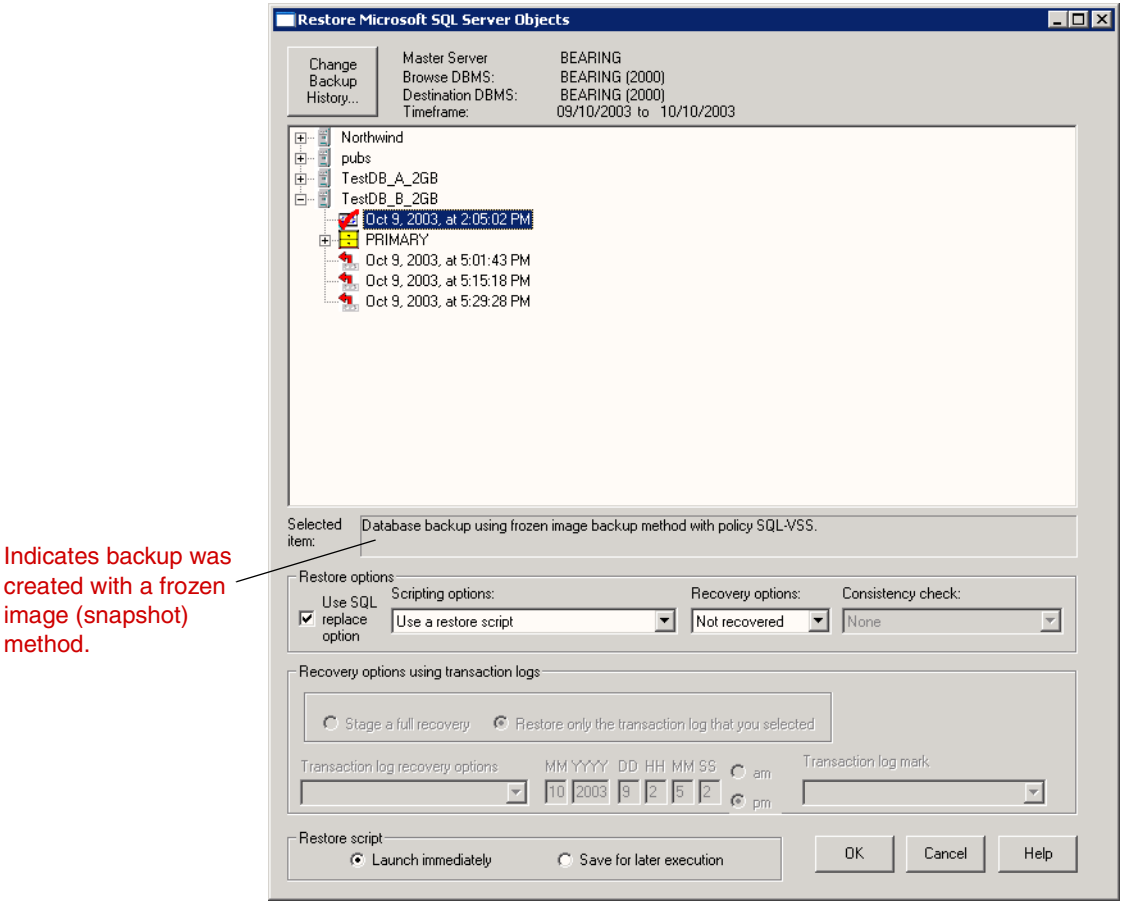
Performing SQL Server Snapshot Backups

There are no special interfacing considerations for performing Advanced Client backups of SQL Server. A snapshot backup will be performed if the backup object is a database, a filegroup, or a file and a policy is selected which is configured for Advanced Client. If a differential or transaction log backup is attempted with an Advanced Client backup, then the operation will use the selected policy, but a standard database backup will be performed using the configured storage unit.

Performing SQL Server Snapshot Restores

Backup images that were created from snapshots are displayed on equal footing with standard backup images. That is, all backup items—without regard to method—are displayed in a time-sequenced ordering that respects the composition of the database hierarchy. In addition, no weighting is given in determining an optimal recovery based on

the backup method. However, you can use the "Restore Microsoft SQL Server Objects" dialog (choose **Actions** > **Restore**) to find out what backup method and policy were used when any given SQL Server backup image was created. Simply, select the image in the history window and observe the information displayed in the **Selected item** box.



Indicates backup was created with a frozen image (snapshot) method.

Grouped Backups

The SQL Server agent provides a method in which multiple databases can be quiesced together and split-off to form a single snapshot. This minimizes the usage of system resources if the databases exist on a single volume, because the aggregation of constituent files would use one snapshot volume instead of one per database. The method for aggregating database Advanced Client backups is called backup *grouping*.

When databases are backed up in a group, all of the databases are quiesced simultaneously, and the constituent files of all of the databases are backed up to a single storage image under the same backup id. This means that an import and copy procedure would use only one image in order to export all of the database backups in the group.

Requirements

The following requirements must be met for a grouped backup to be performed. If any of these requirements are not met, a standard backup will be performed.

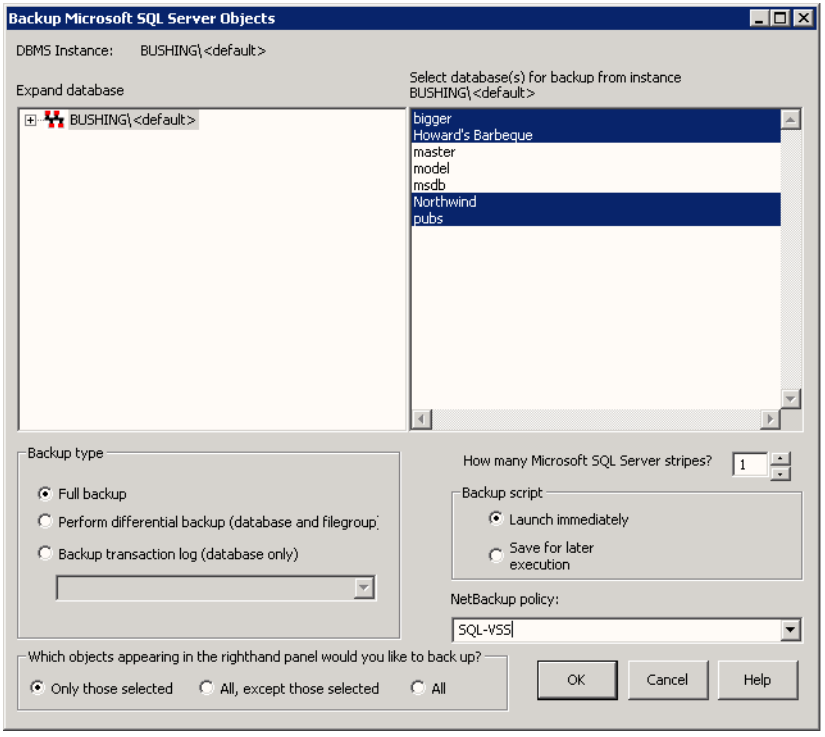
- All backup operations must be full backups. Differential backups are not supported.
- The master database can not be included in a grouped backup.
- The same policy must be specified for each backup operation in the group.
- The same NetBackup server must be specified for each backup operation in the group.

The simplest way to use grouped backup is to select multiple databases using the Backup Microsoft SQL Server Objects dialog box. If the above conditions apply, then the selected databases will be backed up as a group.

Performing a Grouped Backup

▼ To perform a grouped backup

1. Choose (**Actions** > **Backup**).
2. Select the databases you wish to back up.



3. Click **OK**.

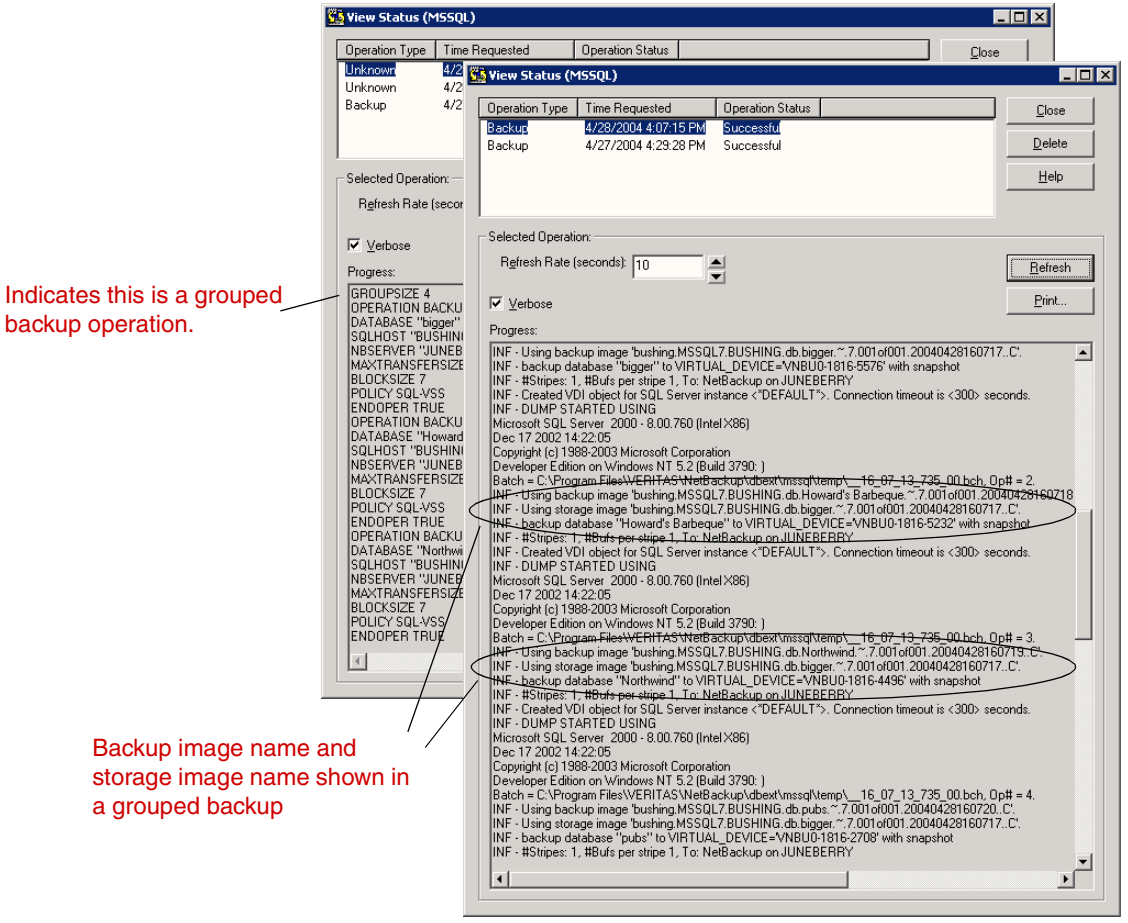
   The backup is launched.

4. To view the progress of the backup, choose **View** > **Monitor Jobs**.

   The batch file used to perform the backup is displayed in the bottom half of the View Status dialog. In the figure "Progress report for a grouped backup operation", notice that the keyword GROUPSIZE appears at the beginning of the batch file. This means that NetBackup will attempt to use grouping to back up the selected SQL Server databases. If the appropriate conditions apply, i.e., all operations are full database

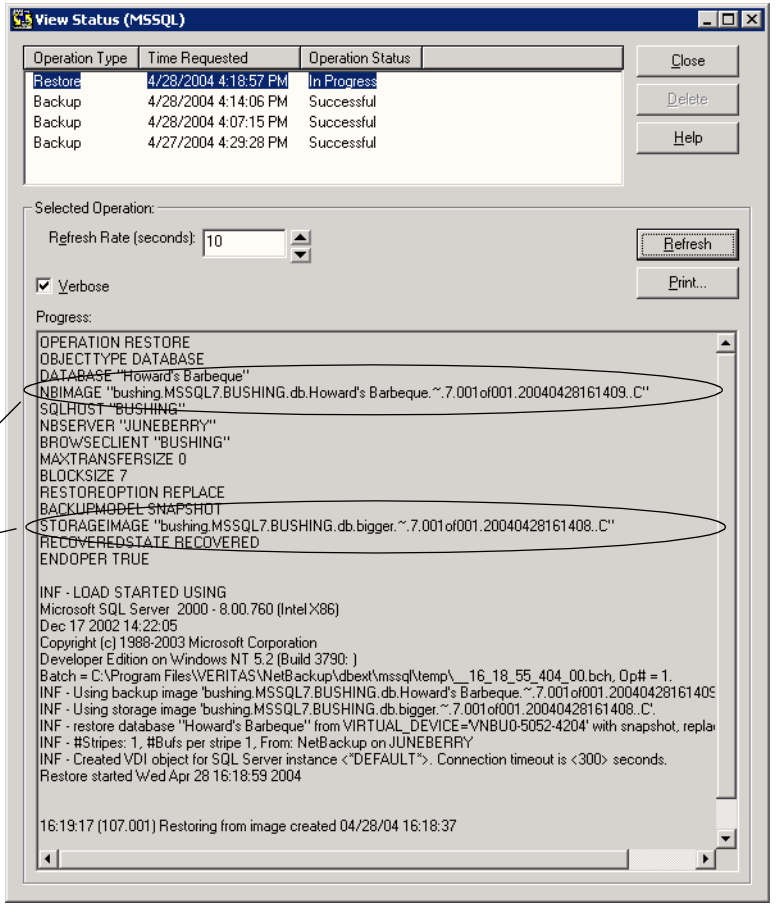
backups, etc., as described above, then all of the databases will be snapped and backed up as a group. When this happens, the progress log will display the backup image name as well as the storage image for each database in the group.

Progress report for a grouped backup operation



Restoring a Database Backed Up in a Group

A database backed up in a group can be restored like any other database. Follow the instructions for "Restoring a Database Backup" on page 76.When you launch the restore operation, notice that the batch file (as displayed below in the progress log monitor) specifies both the name of the storage image and the name of the backup image.



Storage image name and backup image name are shown when restoring from a grouped backup.

Using a Cloaked Database Snapshot to Impact How Differentials are Based

SQL Server records the history of successful database backups in the MSDB system database. It uses this history in deciding how to base differential backups. In particular, SQL Server creates differential database backups as cumulative with respect to the last full database backup that it has recorded in the MSDB. This allows for a quick recovery in case a failure has been detected after the last full database backup.

For example, assume that full backups are created every day at midnight; differentials are created every day at 6AM, noon, and 6PM; and transaction log backups are created every two hours. If a failure occurs at 7:50PM on Tuesday, then a point in time recovery could be achieved by restoring the full database from Tuesday at 12AM, followed by the differential at 6PM on Tuesday, and finally the transaction log at 8PM (specifying "TO 7:50 PM").

However, in some situations with persistent frozen image backups, it may not be feasible to retain the daily full backup after the next full backup has been created. So, if a point in time restore is required prior to the latest backup, the differentials would effectively be based on backups that no longer exist. Thus the only alternative would be to recover based upon the last full backup that had been retained using a potentially long sequence of transaction log images.

To resolve this issue, NetBackup allows you to create SQL Server Advanced Client backups that are not recorded in the MSDB. These are termed *cloaked backups* because SQL Server effectively does not know about them.

As the backup is completing, NetBackup provides SQL Server with an "unsuccessful" status thus inhibiting SQL Server from recording that the backup had occurred. Due to the unsuccessful status, the job line of the MSSQL progress monitor will show a `-1` status, whereas the NetBackup server job monitor registers the job as successful. NetBackup places information in its catalog to denote when a backup has been cloaked and uses this information in determining full database recovery paths.

Sample Backup Schedule Using Cloaked Backups

To understand how recovery staging works with cloaked backups, consider a sample backup schedule with the following characteristics:

- The transaction log is backed up frequently, e.g., every two hours
- A full backup is saved to secondary storage, such as tape, once every several days
- Differential database backups are created several times per day
- A persistent frozen image backup is created several times per day and expires when the next one is created. This backup is cloaked.

The following is an excerpt from this schedule:

Sample backup schedule using cloaked backups

Time	Full backup saved to secondary storage	Differential backup	PFI Cloaked Backup	Transaction log backup
Day 1				
12 AM	✔			✔
2 AM				✔
4 AM		✔		✔
6 AM			✔	✔
8 AM				✔
10 AM		✔		✔
12 PM			✔	✔
2 PM				✔
4 PM		✔		✔
6 PM			✔	✔
8 PM				✔
10 PM		✔		✔
Day 2				
12 AM			✔	✔
2 AM				✔

Under this schedule, full backups are performed every six hours. If a failure occurs, and is detected immediately, then the recovery can be achieved by restoring the last full backup and replaying, on average, three hours of transaction logs. However, if a failure is not detected until after the next full backup, then there would not be any full backups available since 12 AM on day 1. Since the persistent frozen image backups are cloaked, however, the differential backups would each be cumulative with respect to the last non-cloaked full backup.

In this example, suppose that an error occurs at 11:30 PM on day 1 but is not detected until 12:30 AM on day 2, after the 12:AM full backup. Since the 6 PM full backup no longer exists it would be necessary to begin the recovery with the backup taken at 12 AM on day 1. However, since all of the full backups since then were cloaked, the differential backup from 10 PM would be cumulative with respect to that backup. This means that the recovery sequence would be to restore the 12 AM day 1 backup followed by the 10 PM differential backup, followed by 1½ hours of transaction log backups.

If you use cloaked backups, then the "cloaked" attribute is displayed in the Selected Item frame of the Restore SQL Server Objects dialog. Differential backups are automatically associated with the correct full backup and are recognized by the SQL agent when it selects the recovery set for the full database restore.

Caution Microsoft SQL Server does not recognize the cloaked backup. Therefore, if it is incorporated in your database protection strategy, it is essential that you maintain a comprehensive set of transaction logs that span the time duration back to the last non-cloaked full backup.

Creating a Batch File for a Cloaked Backup

▼ To create a batch file for a cloaked backup

1. Open an existing batch file in a text editor.
2. Insert the following:

   ```
   CLOAKEDBACKUP TRUE
   ```

3. Save the batch file.

Configuring a Snapshot Backup

Before configuring NetBackup for SQL Server for snapshot backups, review the configuration requirements and configuration steps in the following two sections. Instructions for configuring snapshot policies are covered in "Configuring an Advanced Client Policy" on page 173 and "Configuring a Policy for Instant Recovery" on page 175.

Configuration Requirements

- See the *NetBackup Advanced Client System Administrator's Guide* for details on the hardware and software requirements for the snapshot method that you want to use. See the VERITAS Technical Support Web site for details on the snapshot methods and platforms that are supported for NetBackup for SQL Server (instructions for accessing this information provided in "Supported Platforms" on page 161).
- It is recommended that the volume(s) which contains the SQL Server databases and log files should be dedicated to SQL Server only. Other types of databases (e.g., Exchange) should not reside on the volume(s).
- NetBackup Advanced Client is installed and configured correctly and the license key for this option has been registered. Refer to the *NetBackup Advanced Client System Administrator's Guide* for details.
- SQL Server 2000 or higher

Configuration Steps

In order to perform an Advanced Client for SQL Server backup, you must perform the following configuration steps:

- ❑ Create a backup script (`.bch` file) using the NetBackup for SQL Server interface. See "Using Batch Files" on page 101.
- ❑ Configure an MS-SQL-Server policy that has the Advanced Client attributes selected. See "Configuring Advanced Client Policies for NetBackup for SQL Server" on page 173.

Configuring Advanced Client Policies for NetBackup for SQL Server

This section only covers what is necessary to configure snapshot backups for a MS-SQL-Server policy. For information on other policy attributes, creating schedules, adding clients, and adding backup selections, see the "Configuration" chapter earlier in this manual. For information on how a snapshot method is selected automatically and details on the types of snapshot methods, refer to the *NetBackup Advanced Client System Administrator's Guide*.

Only one snapshot method can be configured per policy. If, for instance, you want to select one snapshot method for clients one group of clients and a different method for another group then you would need to create two policies for each group of clients and select one method for each policy.

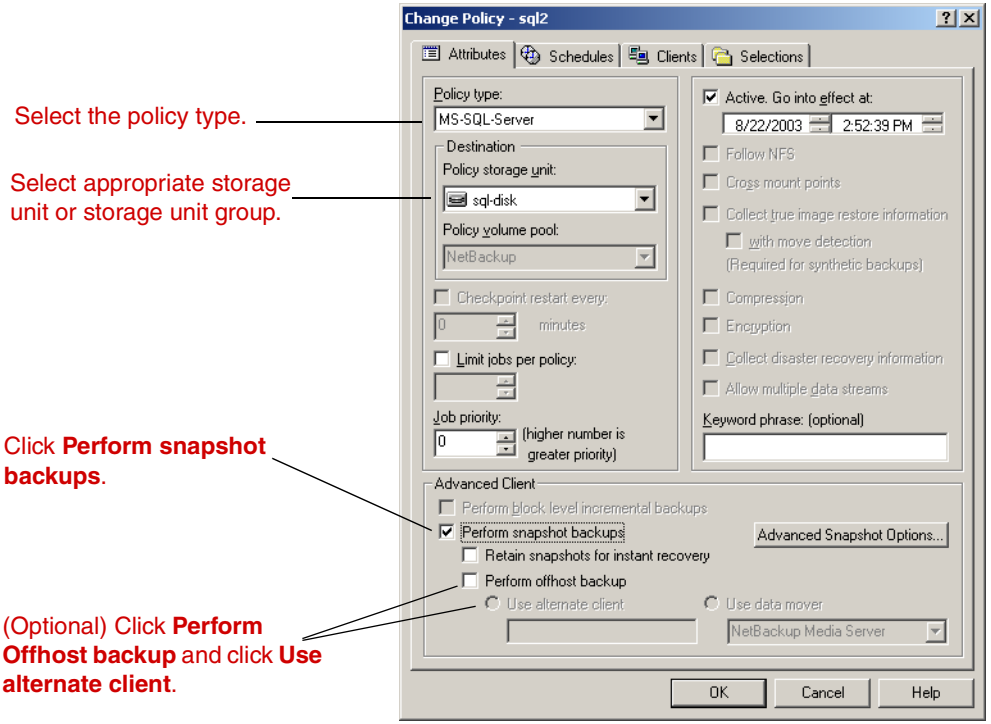
- a snapshot backup (with the option of using an alternate client)
- an instant recovery backup (to disk and to tape, or, to disk only)

Configuring an Advanced Client Policy

This section describes how to configure an Advanced Client policy. Optionally you can choose to have an alternate client perform the backup.

▼ To configure an Advanced Client policy

1. Open the policy you wish to configure.
2. Click on the **Attributes** tab.
3. Select the MS-SQL-Server policy type.



4. Select a **Policy storage unit**.

   If database differentials, filegroup differentials, or transaction logs are included in the Backup Selections list of an Advanced Client policy, then NetBackup will perform a stream-based backup using the selected storage unit. If a storage unit is not provided, then NetBackup will use the default storage unit for the server.

5. Click **Perform snapshot backups**.

6. Choose to have NetBackup select the snapshot method or select the snapshot method manually.

   - By default, NetBackup will choose a snapshot method for you. If you have changed this setting and want NetBackup to choose the method automatically, click **Advanced Snapshot Options** and from the **Snapshot method for this policy** list, choose **auto**.
   - If you wish to use a specific snapshot method, click **Advanced Snapshot Options** and from the **Snapshot method for this policy** list, choose the method you wish to use for this policy.

   See the *NetBackup Advanced Client System Administrator's Guide for* details selecting the snapshot method and automatic snapshot selection.

7. To configure schedules, click on the **Schedules** tab.

   Follow the instructions in the "Configuration" chapter earlier in this manual to configure an Application and Automatic schedule.

8. Optional: If you wish to reduce the processing load on the client, perform the following steps:

   a. The alternate client must be the client that shares the disk array. This option may require additional configuration. Refer to the *NetBackup Advanced Client System Administrator's Guide*.

   b. Select **Perform offhost backup**.

   c. Click **Use alternate client** and enter the name of the alternate client.

Note **Use data mover** is not a supported option for NetBackup for SQL Server.

9. Use the **Clients** tab to specify clients to be backed up by this policy.

   Refer to the "Configuration" chapter earlier in this manual for details on adding clients to a policy.

   Note that for NetBackup 5.0, it is no longer possible to configure a snapshot method on a per-policy, per client basis. Snapshot options are applied to all clients controlled by that policy.

10. Use the **Backup Selections** tab to enter the batch files.

11. Configure other attributes, and add the desired schedules and backup selections. Refer to the "Configuration" chapter earlier in this manual for more information.

12. Click **OK** to close the dialog.

Configuring a Policy for Instant Recovery

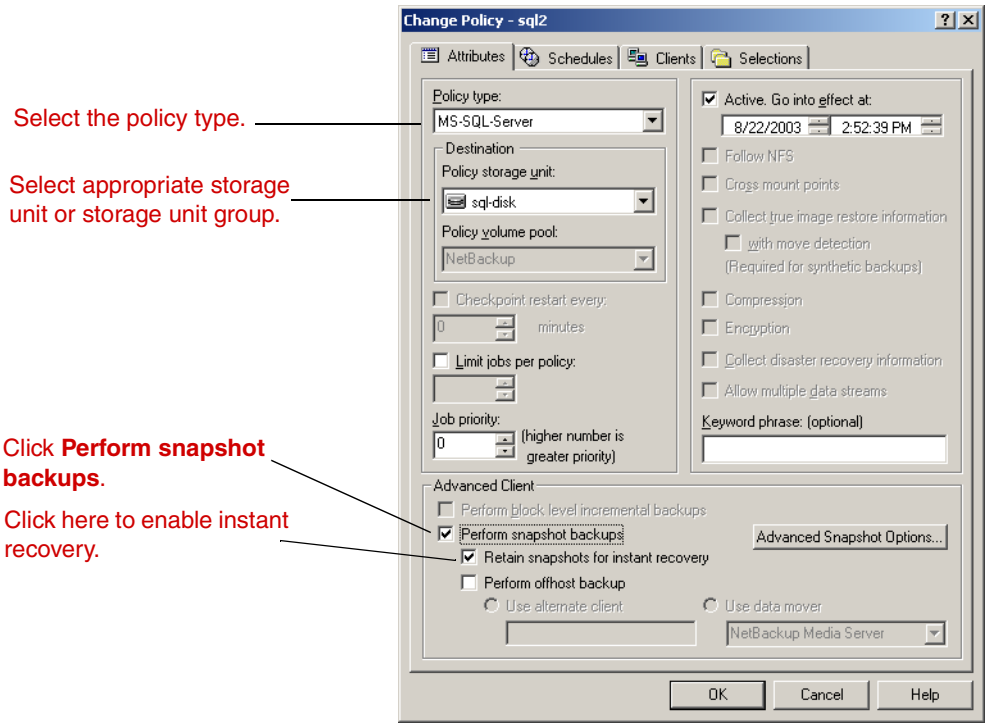
This section describes how to configure a policy for instant recovery. Optionally you can choose to back up to disk only.

▼ To configure a policy for instant recovery

1. Open the policy you wish to configure.

2. Click on the **Attributes** tab.

3. Select the MS-SQL-Server policy type.



4. Select a **Policy storage unit**.

   If you select **Instant recovery backups to disk only** on the **Schedules** tab (see step 8), the storage unit is not used; NetBackup creates only a disk snapshot.

   If database differentials, filegroup differentials, or transaction logs are included in the Backup Selections list of an Advanced Client policy, then NetBackup will perform a stream-based backup using the selected storage unit. If a storage unit is not provided, then NetBackup will use the default storage unit for the server.

5. Click **Perform snapshot backups**.

6. Choose to have NetBackup select the snapshot method or select the snapshot method manually.

   - By default, NetBackup will choose a snapshot method for you. If you have changed this setting and want NetBackup to choose the method automatically, click **Advanced Snapshot Options** and from the **Snapshot method for this policy** list, choose **auto**.

- If you wish to use a specific snapshot method, click **Advanced Snapshot Options** and from the **Snapshot method for this policy** list, choose the method you wish to use for this policy.

See the *NetBackup Advanced Client System Administrator's Guide for* details selecting the snapshot method and automatic snapshot selection.

7. Select **Retain snapshots for instant recovery**.

   NetBackup retains the snapshot on disk, so that instant recovery can be performed from the snapshot. A normal backup to storage is also performed, if you do not select **Instant recovery backups to disk only** (see step 8).

8. To configure schedules, click on the **Schedules** tab.

   a. Follow the instructions in the "Configuration" chapter earlier in this manual to configure an Application and Automatic schedule.

   b. Optional: If you wish to create a disk image only, open the Application schedule and select **Instant recovery backups to disk only**.

      - If **Disk-only backup** is selected, the image is not backed up to tape or other storage. NetBackup creates a disk snapshot only. Note that this disk snapshot is not considered a replacement for traditional backup.
      - If **Disk-only backup** is not selected, NetBackup creates a disk snapshot and backs up the client's data to the storage unit specified for the policy.

Note The **Policy storage unit** is ignored when making a disk image only.

9. Use the **Clients** tab to specify clients to be backed up by this policy.

   Refer to the "Configuration" chapter earlier in this manual for details on adding clients to a policy.

   Note that for NetBackup 5.0, it is no longer possible to configure a snapshot method on a per-policy, per client basis. Snapshot options are applied to all clients controlled by that policy.

10. Use the **Backup Selections** tab to enter the batch files.

11. Configure other attributes, and add the desired schedules and backup selections. Refer to the "Configuration" chapter earlier in this manual for more information.

12. Click **OK** to close the dialog.

Using NetBackup to Restore SQL Server from Backup Exec Images 10

NetBackup can restore SQL Server from Backup Exec images, using the NetBackup Backup, Archive, and Restore (NetBackup Client) interface.

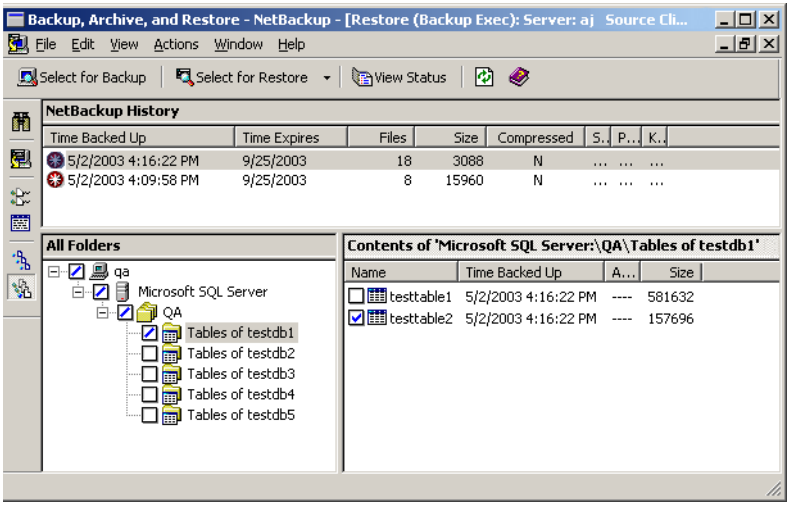
Requirements for Restoring Backup Exec Images

- In order to successfully restore Backup Exec images, the Backup Exec media needs to be imported into NetBackup by first running `vmphyinv`, and then running `bpimport`, Phase 1 and Phase 2. Refer to the instructions for importing images from Backup Exec media in the *NetBackup System Administrator's Guide for Windows, Volume I*. NetBackup can restore Backup Exec images of SQL 7.0 and SQL 2000.
- The NetBackup Client Service should be running in a user account that has been granted the System Administrator role on the target SQL instance. See "DBMS (Database Management System) Privileges" on page 46 for details on configuring the logon account for this service.

Limitations When Restoring Backup Exec Images

- Restoring multiple databases in a single restore job is not supported. It is however possible to restore full, differential and transaction log backups of a same database in a single restore job.
- Table backups cannot be restored, though they are included in the backup image and can be selected for restore (See the figure below). If you attempt to restore a table, the job will fail with the error "unsupported object was selected for restore".

Table backups cannot be restored with NetBackup.



Restoring from Backup Exec Images

This section describes the restore options available and the procedures for restoring SQL Server from Backup Exec images.

Restore Options

The following options are available when restoring from SQL backups from Backup Exec images.

Restore options on the Microsoft SQL Server tab

Item	Description
Recovery completion state	
Leave database operational. No additional transaction logs can be restored. (With Recovery)	Select this option when restoring the last database, differential, or log backup in the restore sequence in order to have the restore operation roll back all uncompleted transactions. After the recovery operation, the database is ready for use. If Leave database operational is not performed, the database is left in an intermediate state and is not usable. If Leave database operational is selected when an intermediate backup is being applied, you cannot continue to restore backups. You must restart the restore operation from the beginning.
Leave database nonoperational but able to restore additional transaction logs (No Recovery)	Select this option during a restore if you have additional differential or transaction log backups to be restored in another restore job.
Leave database read-only and able to restore additional transaction logs (Standby)	Select this option during transaction log and database restore to create and maintain a standby database. See your SQL documentation for information on standby databases.

Restore options on the Microsoft SQL Server tab

Item	Description
Replace databases or filegroups	Select this checkbox to replace a database or filegroup, even if another database or filegroup with the same name already exists on the server. If Replace Databases or Filegroups is not specified for a restore, SQL performs a safety check to ensure that a different database or filegroup is not accidentally overwritten. Refer to your SQL documentation for more information about the safety check that occurs when the REPLACE option is not selected.
Automate master database restore	Select this checkbox to enable NetBackup to stop SQL so that the master database can be restored. All existing users are logged off, and SQL Server is put into single-user mode. When this option is selected, only the master database can be restored; if this option is selected for any other database, those jobs will fail. If NetBackup does not have access to the SQL registry keys, HKEY_LOCAL_MACHINE\Software\Microsoft\ Microsoft SQL Server, and HKEY_LOCAL_MACHINE\Software\Microsoft\MSSQLServer, then a restore to the default directory may not work, and the option **Automate master database restore** will not work. To ensure that NetBackup has access rights, verify that the user account of the NetBackup client service has administrator rights to the Windows server that the SQL instance is installed on.
Consistency check after restore	• **Full check, excluding indexes**. Select this to exclude indexes from the consistency check. If indexes are not checked, the consistency check runs significantly faster but is not as thorough. Only the data pages and clustered index pages for each user table are included in the consistency check. The consistency of the nonclustered index pages is not checked. • **Full check, including indexes**. Select this to include indexes in the consistency check. Any errors are logged. This option is selected by default. • **Physical check only (SQL 2000 only)**. Select this to perform a low overhead check of the physical consistency of the SQL 2000 database. This option only checks the integrity of the physical structure of the page and record headers, and the consistency between the pages' object ID and index ID and the allocation structures.

Restore options on the Microsoft SQL Server tab

Item	Description
	• **None**. Select this if you are doing sequential restores. Do not run a consistency check after a restore until all sequential restores have been done. If a consistency check is selected during a restore, the restore will complete but the consistency check will not be done. Check the job log for this information. If you need to recover the database after restores are complete, select one of the following consistency checks when you select the **Leave database operational** option.
Note Do not select the **Alternate drive** option when restoring filegroups. Filegroups must be restored to the same drive letter and path that they were backed up from.	
Alternate drive for restoring database files	Use this option to select a drive to which SQL database files can be restored if the drive where one or more of the database files previously resided no longer exists. When a SQL database is backed up, the physical file names (which include the directory path) of the files that make up the database are stored in the backup set by SQL. For example, for the logical file pubs, the physical file name is stored as E:\MSSQL7\DATA\pubs.mdf. If the database must later be restored, SQL uses these same physical file names to target the restore to. During a restore, NetBackup automatically creates any necessary subdirectories that do not exist. However, if the drive where one or more of the database files previously resided no longer exists, NetBackup moves those files to their original directory path, but on the alternate drive specified. Using the same example, if drive C: is specified, then the file with the original directory path of E:\MSSQL7\DATA\pubs.mdf is restored to C:\MSSQL7\DATA\pubs.mdf. If no alternate drive is specified in this situation, the job will fail.
Restore to alternate drive	
Only when original drive does not exist	Select this option to restore all database files to their original directory path on the alternate drive, only if the drive where they originally resided exists. To make this option available, select a drive letter in **Alternate drive for restoring database files** list.
Even when original drive does exist	Select this option to restore all database files to their original directory path on the alternate drive, even if the drive where they originally resided exists. To make this option available, select a drive letter in **Alternate drive for restoring database files** list.

Restore options on the Microsoft SQL Server tab

Item	Description
Note Do not select the **Restore all database files to the target instance's data location** option when restoring filegroups. Filegroups must be restored to the same drive letter and path that they were backed up from.	
Restore all database files to the target instance's data location	Select this checkbox to restore files to the default data and log directories of the destination instance. For example, if you are restoring a database to a different instance of SQL, you would select this option to move the database files to the correct location for the new instance. If this option is not selected, then the files are restored to the directory that the master database is in.
Point in time log restore	Select this checkbox to restore transactions from a transaction log up to and including a point in time in the transaction log. After the point in time, recovery from the transaction log is stopped. Select the part of the date you want to change, and then enter a new date or click the arrow to display a calendar from which you can select a date. Select the part of the time you want to change, and then enter a new time or click the arrows to select a new time.
Restore log up to named transaction (SQL 2000)	Select this checkbox to restore transactions from a transaction log up to a named transaction (or named mark) in the transaction log; after that, recovery from the transaction log is stopped. The named transactions are case-sensitive.
Include the named transaction	Select this checkbox to include the named transaction in the restore; otherwise the restore will stop immediately before the named transaction is restored.
Found after	Select this checkbox to specify a date and time after which the restore operation is to search for the named transaction. For example, if you specify a restore from a log up to the named transaction *AfternoonBreak*, found after 6/02/2003, 12:01 p.m., then the restore operation will not search for *AfternoonBreak* until after that time.

Restore Options for Redirected Restores

Restores options on the Microsoft SQL Server Redirection tab

Item	Description
Redirect Microsoft SQL Server sets	Select this checkbox to enable redirection of SQL backup sets.
SQL Logon Account User Name	If SQL Server Authentication is being used, enter the logon account that stores the credentials of the SQL user account. You do not need to provide a user name and password if integrated security is being used. See "DBMS (Database Management System) Privileges" on page 46 for more information on standard and integrated security.
SQL Logon Account Password	Indicate the password associated with logon account.
SQL Redirection	To redirect this restore to a named instance, type the name of the server and the instance name, in the format "server_name instance_name". For example, **`aj civolus`**. If you are redirecting the default instance, type the name of the server, in the format "server_name". For example, **aj**.
Restore to Database	To redirect the restore to a different database on the target server, type the target database name; otherwise, leave the field blank. You can redirect a full database backup to a different server and/or database; however, if the drive configuration is different from when the database backup was created, you must select either **Alternate drive for restoring database files** or select **Restore all database files to the target instance's data location** on the Microsoft SQL Server tab. See "Restore Options" on page 181. If you are restoring a differential or log backup, and the associated database backup was restored to a different server, enter the new database name.

Specifying the Server, Client, and Policy Type

In order to browse for backups of SQL databases backed up by Backup Exec, you must first specify the server containing the backup images, the client that performed the backups, and the type of policy associated with the backups.

▼ To choose the server, client, and policy type

1. Choose **File** > **Specify NetBackup Machines and Policy Type**.

   The Specify NetBackup Machines and Policy Type dialog is displayed.

2. Click the **Servers** tab.

3. Verify that the server on which the backup images are stored is marked as CURRENT.

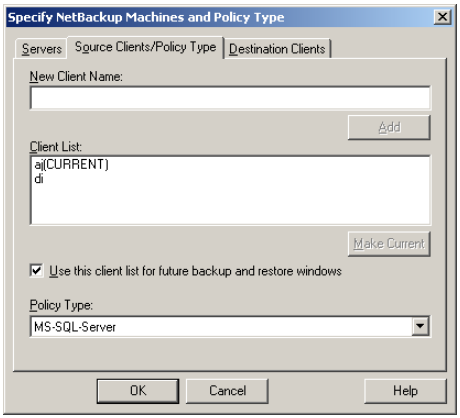
4. Click the **Source Clients/Policy Type** tab.

5. Verify that the client whose images need to be restored is marked as CURRENT.

6. If you wish to restore to a different client, click the **Destination Clients** tab and select the desired client.

   See "Redirecting a Restore" on page 199 for additional information on redirecting a Backup Exec restore.

7. From the **Policy Type** list, select **MS-SQL-Server**.



8. Click **OK**.

   NetBackup browses for SQL Server backup images.

   The NetBackup History pane displays SQL Server backup information. The top split windows show individual image information and the bottom split gives file and folder information and also allows the user to select what files are to be restored.

Restoring Backup Exec Database Backups

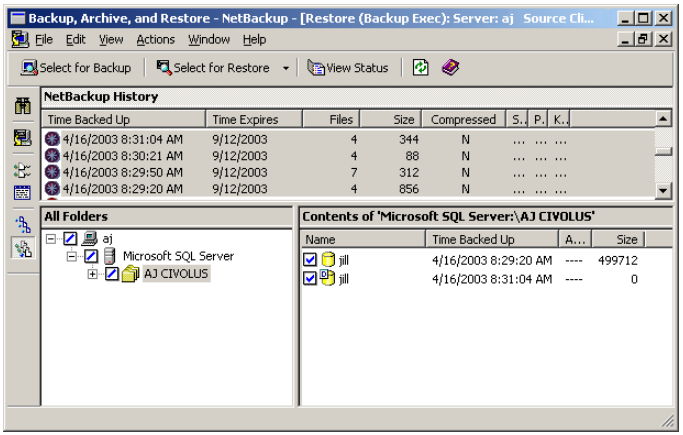
If the database you wish to restore is using the simple recovery model, there are no transaction log backups to restore. You only need to restore the most recent full database backup and if you were running differential database backups, restore the most recent differential database backup.

▼ To restore a database backup

1. Log on as Administrator.
2. Open the Backup, Archive, and Restore interface.
3. Specify the appropriate server, client, and policy type, as described in "Specifying the Server, Client, and Policy Type" on page 186.
4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.
5. From the NetBackup History pane, select the backup image containing the objects you wish to restore.

6. Select the most recent full database backup, and the most recent differential database backup, if any, to restore.

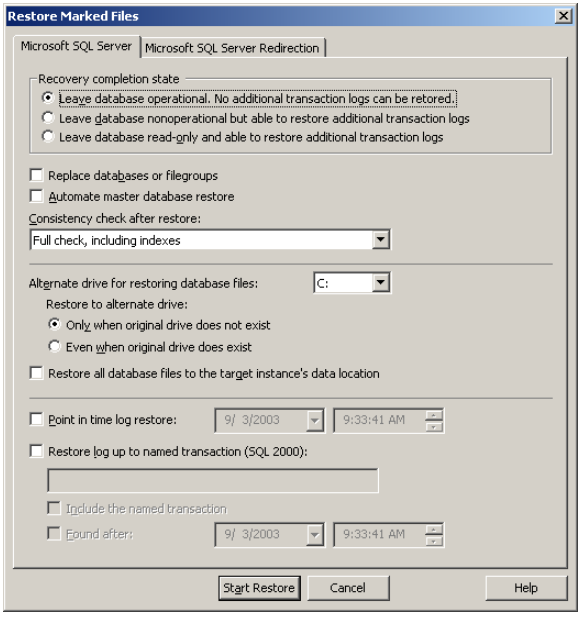


7. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

8. From the Recovery completion state group, choose **Leave database operational. No additional transaction logs can be restored.**

For information on other SQL restore options, refer to "Restore Options" on page 181.



9. Click **Start Restore**.

Restoring Backup Exec SQL Transaction Logs Backups Up to a Point in Time

You can restore transactions from a transaction log up to and including a point in time in the transaction log. After the point in time is reached, recovery from the transaction log is stopped.

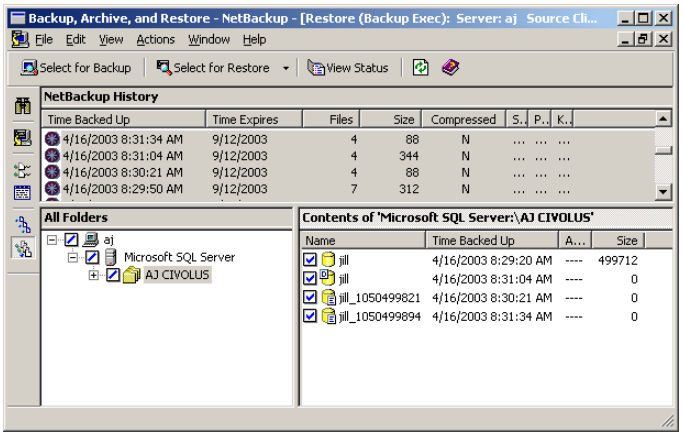
▼ To restore SQL transaction logs up to a point in time

1. Log on as Administrator.
2. Open the Backup, Archive, and Restore interface.
3. Specify the appropriate server, client, and policy type, as described in "Specifying the Server, Client, and Policy Type" on page 186.
4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.

5. From the NetBackup History pane, select the backup image containing the objects you wish to restore.

6. In the All Folders pane, select the most recent full database backup, and the most recent differential database backup, if any, and all the log backups you want to restore.



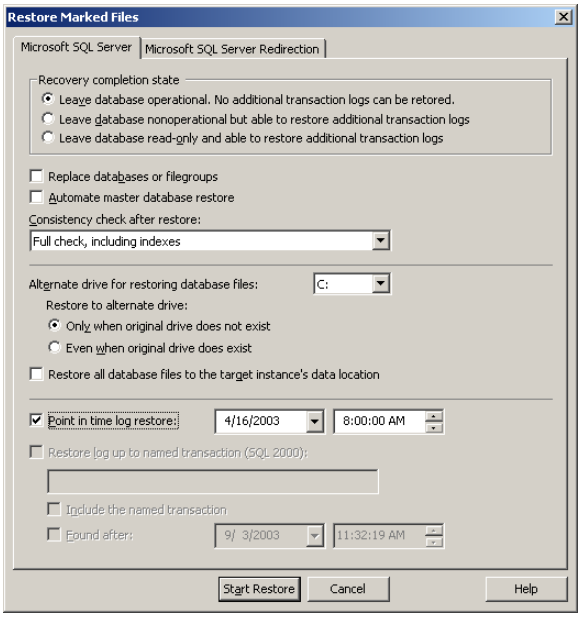
7. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

8. From the Recovery completion state group, choose **Leave database operational. No additional transaction logs can be restored.**

9. Select **Point in time log restore**, and then select a date and time.

For information on other SQL restore options, refer to "Restore Options" on page 181.



10. Click **Start Restore**.

Restoring Backup Exec SQL Transaction Logs Backups Up to a Named Transaction

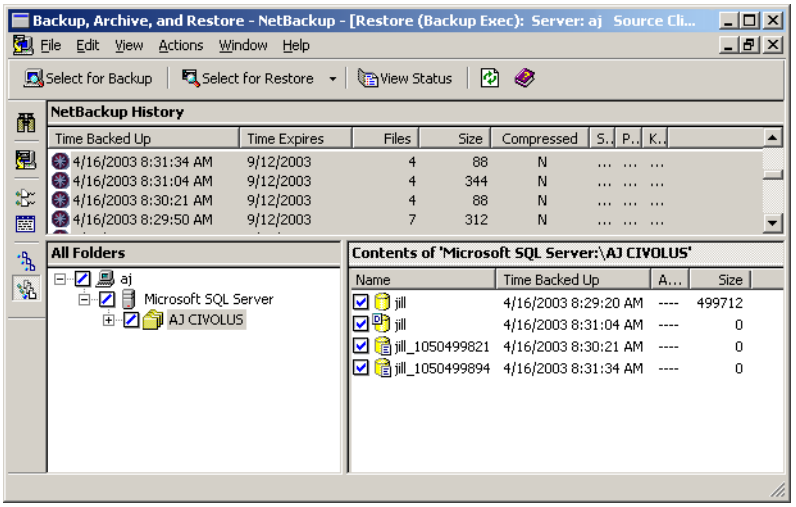
When restoring SQL 2000 transaction logs, you can restore transactions from a transaction log up to and including a named transaction (or mark). After the named transaction is reached, recovery from the transaction log is stopped.

▼ To restore a transaction log up to a named transaction

1. Log on as Administrator.
2. Open the Backup, Archive, and Restore interface.
3. Specify the appropriate server, client, and policy type, as described in "Specifying the Server, Client, and Policy Type" on page 186.
4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.

5. From the NetBackup History pane, select the backup image containing the objects you In the All Folders pane, select the most recent full database backup, and the most recent differential database backup, if any, and all the log backups you want to restore.



6. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

7. From the Recovery completion state group, choose **Leave database operational. No additional transaction logs can be restored.**

8. Select **Restore log up to named transaction**, and then enter the name of the transaction.

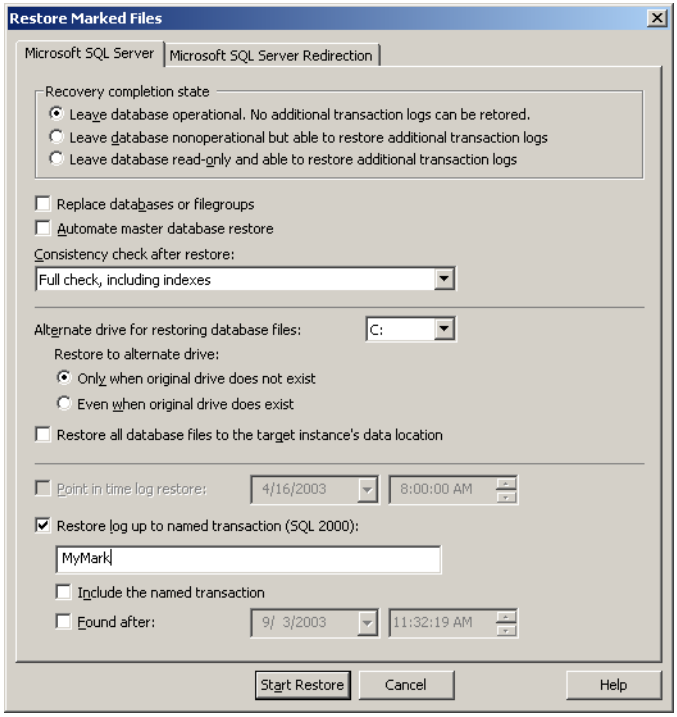
   The names are case-sensitive. Make sure you enter the correct upper- and lower-case characters.

9. To include the named transaction in the restore, select **Include the named transaction**.

10. To specify a particular named transaction in the log, select **Found after** and then select a date and time.

    If a date and time are not entered, recovery from the transaction log is stopped at the first transaction with the specified name.

For information on other SQL restore options, refer to "Restore Options" on page 181.



11. Click **Start Restore**.

Restoring Backup Exec Filegroup Backups

With filegroup backups, you can restore the entire database, a primary filegroup, a filegroup containing a deleted or changed table, and a nonprimary filegroup.

▼ To restore a filegroup

Note Use separate restore jobs to restore the primary filegroup, the rest of the filegroup backup sets, and the transaction logs.

1. Log on as Administrator.

2. Open the Backup, Archive, and Restore interface.

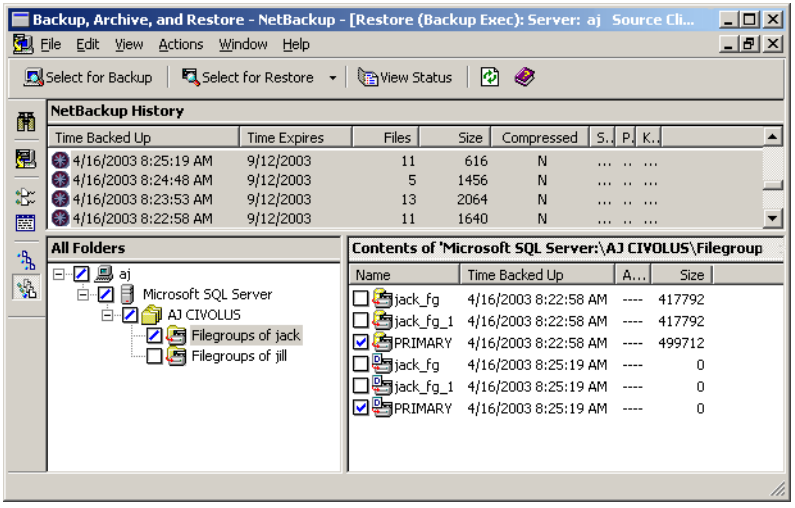
3. Specify the appropriate server, client, and policy type, as described in "Specifying the Server, Client, and Policy Type" on page 186.

4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.

5. From the NetBackup History pane, select the backup image containing the objects you wish to restore.

6. In the All Folders pane, select the full backup of the primary filegroup and any differential backups.

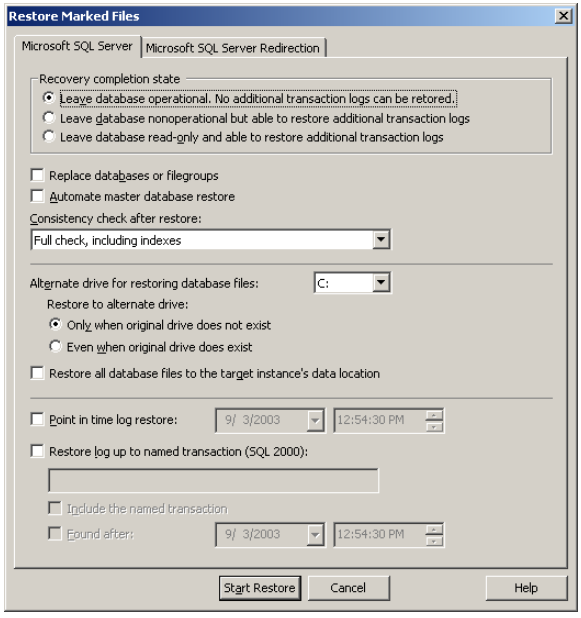


7. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

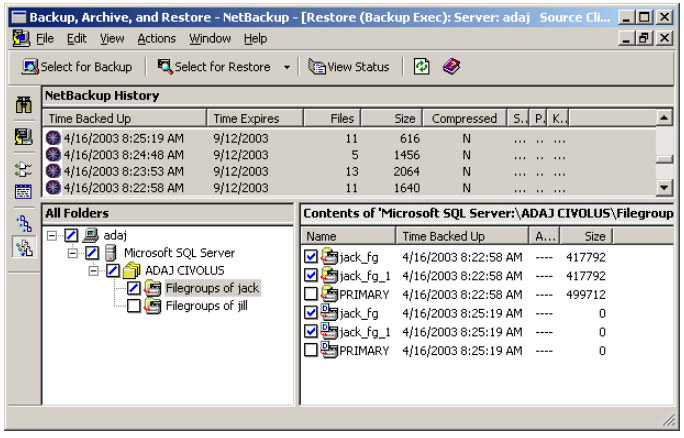
8. From the Recovery completion state group, choose **Leave database nonoperational but able to restore additional transaction logs**.

For information on other SQL restore options, refer to "Restore Options" on page 181.

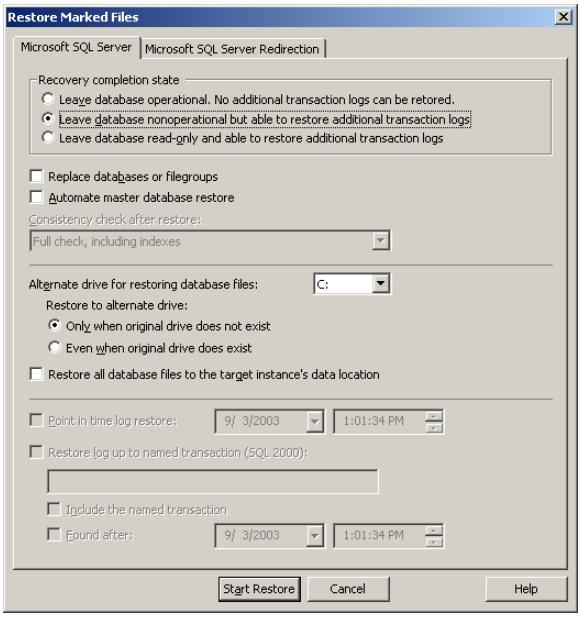


9. Click **Start Restore**.

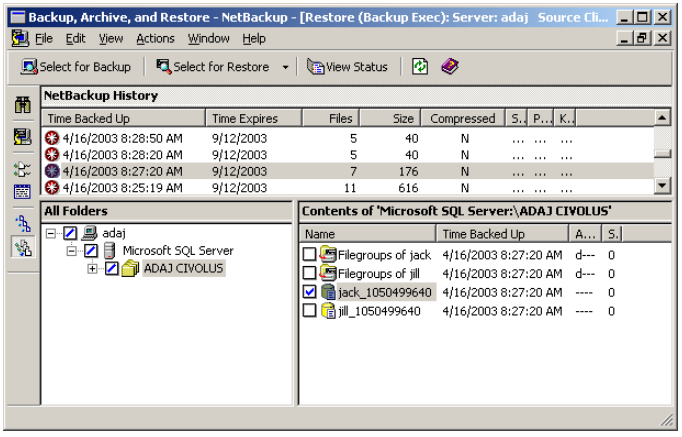
10. After the primary filegroup is restored, select the latest full and differential backups for the other filegroups.



11. On the Microsoft SQL Server tab, select the recovery completion state **Leave database nonoperational but able to restore additional transaction logs**, and then start the restore job.



12. When the other filegroups are restored, select the transaction logs.



13. On the Microsoft SQL Server tab, select the recovery completion state **Leave database operational and restore all of the transaction logs**, or select **Point in time log restore** or **Restore log up to named transaction**.

14. Click **Start Restore**.

Restoring the SQL Master Database from a Backup Exec Image

If the master database is damaged, symptoms may include:

- An inability to start SQL.
- Segmentation faults or input/output errors.
- A report generated by SQL Database Consistency Checker utility (DBCC).

If you can still start SQL, you can restore the latest copy of the master database backup using the **Automate master database restore** option on the **Microsoft SQL Server tab** and then restore any other databases, if needed.

If the master database is critically damaged and SQL cannot be started, rather than running the Rebuild Master utility or reinstalling SQL to be able to restart SQL, you can replace the corrupted or missing databases with the copies of the master and model databases that Backup Exec automatically creates and updates whenever backups of those databases are run. After SQL is running again, you can restore the latest copy of the master database using the **Automate master database restore** option, and then restore any other databases, if needed.

If copies of the master and model databases were not made, then you must use Microsoft's `rebuildm.exe` utility to rebuild the master database and start SQL.

Because all changes made to the master database after the last backup was created are lost when the backup is restored, the changes must be reapplied. If any user databases were created after the master database was backed up, those databases cannot be accessed until the databases are restored from backups or reattached to SQL.

▼ To restart SQL using database copies:

1. Verify that the database copies are present.

   The database copies are named master$4idr, mastlog$4idr, model$4idr, and modellog$4idr.

   In a default installation of SQL 2000, the databases are in:

   ```
   C:\Program Files\Microsoft SQL Server\MSSQL\Data\*.*.
   ```

   In a named instance of SQL 2000, the databases are in:

   ```
   C:\Program Files\Microsoft SQL Server\MSSQL$Instance_Name\Data\*.*
   ```

   If necessary, restore the master and model database copies from a backup set to the same directory that the original master and model databases are in.

2. Open a command prompt window, and delete the original master and model databases and their transaction logs.

   For example:

   ```
   C:\Program Files\Microsoft SQL Server\MSSQL\Data> del master.mdf
   mastlog.ldf model.mdf modellog.ldf
   ```

3. Rename the copies of the databases back to their original names.

   Type the following:

   ```
   C:\Program Files\Microsoft SQL Server\MSSQL\Data>rename
   master$4idr master.mdf

   C:\Program Files\Microsoft SQL Server\MSSQL\Data>rename
   mastlog$4idr mastlog.ldf

   C:\Program Files\Microsoft SQL Server\MSSQL\Data>rename model$4idr
   model.mdf

   C:\Program Files\Microsoft SQL Server\MSSQL\Data>rename
   modellog$4idr modellog.ldf
   ```

4. Use the SQL Service Control Manager to start SQL Server.

5. Continue with the next procedure to restore the latest changes to the master database.

▼ To restore the master database

1. Log on as Administrator.

2. Open the Backup, Archive, and Restore interface.

3. Specify the appropriate server, client, and policy type, as described in "Specifying the Server, Client, and Policy Type" on page 186.

4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.

5. From the NetBackup History pane, select the backup image containing the last master database backup.

6. In the All Folders pane, select the master database.

7. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

8. On the **Microsoft SQL Server** tab, select **Automate master database restore**.

   When this option is selected, only the master database can be restored; if this option is selected for any other database, those jobs will fail. When you start the restore operation, all existing users are logged off, and SQL Server is put into single-user mode.

   If NetBackup does not have access to the SQL registry keys HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server and HKEY_LOCAL_MACHINE\Software\Microsoft\MSSQLServer, then a restore to the default directory may not work, and the option **Automate master database restore** will not work. To ensure that NetBackup has access rights, verify that the account that NetBackup uses has administrator rights to the computer that is running SQL.

9. Select a consistency check to be run after the restore.

10. Click **Start Restore**.

    After the restore, SQL is restarted in multi-user mode.

Redirecting a Restore

You can redirect the following:

- A database backup to a different server, database, or instance.
- Differential and log backups to wherever the associated database is restored.
- One or more filegroups in a backup to a different server or instance. Filegroups can be redirected to a different server, but the database file paths cannot be changed. For example, if the filegroup was backed up from `G:\SQLDATA`, then it must be restored to `G:\SQLDATA`, even if it is redirected to another server. Filegroups must be restored to the same drive letter and path that they were backed up from.

▼ To redirect a restore

1. Log on as Administrator.

2. Open the Backup, Archive, and Restore interface.

3. Specify the appropriate server, source client, destination client, and policy type, as described in "Specifying the Server, Client, and Policy Type" on page 186.

4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.

5. Choose the objects you wish to restore.

6. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

7. Click on the **Microsoft SQL Server Redirection** tab.

8. Refer to "Restore Options for Redirected Restores."

9. Select the other desired restore options on the **Microsoft SQL Server** tab. For more information, refer to "Restore Options" on page 181.

10. Click **OK**.

Troubleshooting 11

This chapter describes the tools that are available to help you prevent some common, however infrequent, problems encountered with the daily operation of the SQL Server. For more in-depth information, please refer to the Microsoft Support web site.

This chapter contains information on the following troubleshooting tools:

- Progress Reports Created for NetBackup for SQL Server on the Client
- NetBackup for SQL Server Logs
- Changing the Debug Level
- NetBackup Reports
- Preventing Timeout Failures on Large Database Restores

Progress Reports Created for NetBackup for SQL Server on the Client

NetBackup for SQL Server creates a progress report for each operation that has been initiated. This report contains summary information concerning the overall status of your job. The reports are contained in directory *install_path*\NetBackup\logs\user_ops\mssql\logs and can be conveniently viewed by opening the progress report viewer from the NetBackup for SQL Server GUI.

The progress report contains the following types of information:

- the batch keywords and values which define the operation. See "Overview of Batch Files" on page 101 for information about the batch file syntax
- summary information about the operation
- information about the operation as it progresses
- any error conditions or warnings that cause the operation to fail
- the final outcome of the operation, whether it succeeded or failed and how long it took

The following is a typical progress report created for a database backup.

Line	Text
1	OPERATION BACKUP
2	DATABASE "Howard's Famous Barbeque Ribs"
3	SQLHOST "JUY"
4	SQLINSTANCE "NEWINSTANCE"
5	NBSERVER "JUY"
6	MAXTRANSFERSIZE 0
7	BLOCKSIZE 0
8	ENDOPER TRUE
9	INF - DUMP STARTED USING
10	Microsoft SQL Server 2000 - 8.00.760 (Intel X86)

Line	Text
11	Dec 17 2002 14:22:05
12	Copyright (c) 1988-2003 Microsoft Corporation
13	Enterprise Edition on Windows NT 5.0 (Build 2195: Service Pack 3)
14	Batch = D:\Program Files\VERITAS\NetBackup\dbext\mssql\temp__bch12_41_04_088_00.bch, Op# = 1.
15	INF - Using backup image 'juy.MSSQL7.JUY\NEWINSTANCE.db.Howard's Famous Barbeque Ribs.~.0.001of001.20030723124105..C'.
16	INF - backup database “Howard's Famous Barbeque Ribs“ to VIRTUAL_DEVICE='VNBU0-1548-2620' with blocksize = 512, maxtransfersize = 65536, buffercount = 1
17	INF - #Stripes: 1, #Bufs per stripe 1, To: NetBackup on JUY
18	INF - Created VDI object for SQL Server instance <NEWINSTANCE>. Connection timeout is <300> seconds.
19	INF - DBClient has been closed for stripe #0
20	12:41:07 Initiating backup
21	12:41:12 INF - Data socket = juy.domain.com.4146
22	12:41:12 INF - Name socket = juy.domain.com.1527
23	12:41:12 INF - Job id = 143
24	12:41:12 INF - Backup id = juy_1058982070
25	12:41:12 INF - Backup time = 1058982070
26	12:41:12 INF - Policy name = sql
27	12:41:12 INF - Frozen image = 0
28	12:41:12 INF - Backup copy = 0
29	12:41:12 INF - Master server = juy
30	12:41:12 INF - Media server = juy

Line	Text
31	12:41:12 INF - Multiplexing = 0
32	12:41:12 INF - New data socket = juy.domain.com.1388
33	12:41:12 INF - Use shared memory = 1
34	12:41:12 INF - Compression = 0
35	12:41:12 INF - Encrypt = 0
36	12:41:12 INF - Client read timeout = 300
37	12:41:12 INF - Media mount timeout = 0
38	12:41:16 INF - Data buffer size = 262144
39	12:41:18 INF - Beginning backup on server juy of client juy.
40	12:41:25 INF - Server status = 0
41	12:41:26 INF - Backup by hao on client juy using policy sql: the requested operation was successfully completed.

Observe the following:

- Lines 1 to 8 contain the batch syntax which drove this operation. This operation was started from a GUI created batch file.
- Line 9 indicates that this was a backup (“Dump”). The term ‘Load’ would indicate a restore.
- Lines 10-14 provide the version of SQL Server and the Windows operating system.
- Line 15 provides the name of the batch file. Since the backup was launched immediately, it was placed in *`install_path`*`\NetBackup\dbext\mssql\Temp` directory.
- Line 16 provides the actual SQL syntax that was used to launch the backup command.
- Line 17 indicates that this was a single stream backup.
- Line 18 provides the SQL Server instance name and the virtual device interface timeout.
- Lines 21 to 42 provide NetBackup-based client parameters and statuses. Note that the server status on line 40 indicates that the operation completed with status 0, which indicates success.

NetBackup for SQL Server Logs

For detailed troubleshooting information, create the following folders on each NetBackup for SQL Server host. You can disable logging by deleting the folders.

- Create the following folders and allow public access to them:

  install_path\NetBackup\logs\bphdb\
  install_path\NetBackup\logs\dbclient\

Log Descriptions

Log names are formatted as MMDDYY.log. The resulting debug logs created will contain the following types of information. For additional NetBackup client logs and NetBackup master server logs, see the online help for the Backup, Archive, and Restore interface and the *NetBackup System Administrator's Guide for Windows, Volume I* or *NetBackup System Administrator's Guide for UNIX, Volume I*.

bphdb Log

The bphdb log contains bphdb process information. bphdb is the NetBackup Hot Database Backup binary and is the client process that NetBackup for SQL Server uses to start the backup or restore operations schedules from the NetBackup master server logs. The bphdb logs are normally not very long.

Note Use the **Debug Verbose Level** option in the NetBackup Backup, Archive, and Restore interface to adjust the amount of information that appears in the bphdb log. This verbose level applies to all of the NetBackup logs used on the Windows client except for dbclient.

dbclient Log

The dbclient log contains process information from the SQL Server agent and from the VERITAS XBSA module. The XBSA module is the NetBackup client interface program which connects to the NetBackup master server. This log also contains process information for dbbackup and dbbackmain.

Note To specify how much information appears in the dbclient debug log, set the TRACELEVEL parameter to MIN, MID, or MAX. The MAX level will produce huge amounts of output, which is usually appropriate only for internal debugging. The TRACELEVEL parameter is set for individual operations either through the batch file or the NetBackup Database Extension GUI.

Changing the Debug Level

You can control the amount of information written to the debug log in the *install_path*\NetBackup\logs\dbclient folder by changing the General. The higher the value, the more information is logged. In everyday normal operations, the default value of 0 is sufficient. However, VERITAS technical support may ask you to set the value higher when a problem is being analyzed.

You can also control the amount of information in the dbclient debug log. The TRACELEVEL parameter is set for individual operations either through the batch file or in the **Client Options** dialog in the NetBackup Database Extension GUI. To specify how much information appears in the dbclient debug log, set the TRACELEVEL parameter to MIN, MID, or MAX. The MAX level will produce huge amounts of output, which is usually appropriate only for internal debugging.

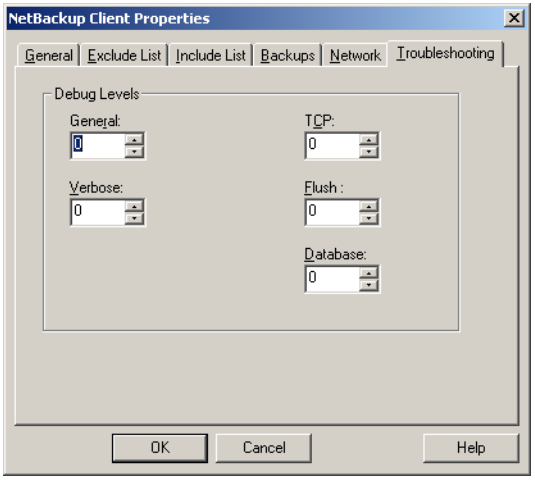
▼ To change the Debug Level

1. Choose **Start** > **Programs** > **VERITAS NetBackup** > **Backup, Archive, and Restore**.

   The Backup, Archive, and Restore - NetBackup window appears.

2. Choose **File** > **NetBackup Client Properties**.

3. Click the **Troubleshooting** tab.



   By default, the settings are zero.

4. Set the **General** to adjust the amount of information written to the NetBackup Server logs.

5. Set the **Database** debug level to adjust the amount of information written to the `dbclient` log.

6. Click **OK** to save your changes.

NetBackup Reports

The administrator has access to operational progress reports through administrator interfaces. Reports may be generated for Backup Status, Client Backups, Problems, All Log Entries, Media Lists, Media Contents, Images on Media, Media Logs, Media Summary, and Media Written. These reports may be generated for a specific time frame, client, or master server. Refer to the NetBackup System Administrator's Guide for UNIX, Volume I or the NetBackup System Administrator's Guide for Windows, Volume I for details.

Preventing Timeout Failures on Large Database Restores

A large SQL Server may fail with a Client Read Timeout error before any data has been read from the NetBackup media. This occurs because the SQL Server may need to pre-write the database files before the restore operation begins. The time required for this process is a function of the size of the database files and the speed at which your host machine can write to disk. For example, if your system can perform disk writes at the rate of 60 megabytes per second and you have a 2.4 terabyte database, then it will take at least 12 hours for SQL Server to prep the disk before the actual restore can begin. In reality, the delay may be even longer than what you calculate by as much as 20% to 40%.

The timeout problem can be resolved by increasing the NetBackup Client Read Timeout setting. Use the NetBackup Administration Console on the server to change the properties of each client that contains a database you may need to restore. The default for the Client Read Timeout setting is 300 seconds (5 minutes). If you have clients which contain large SQL Server databases, you may need to set this value much higher.

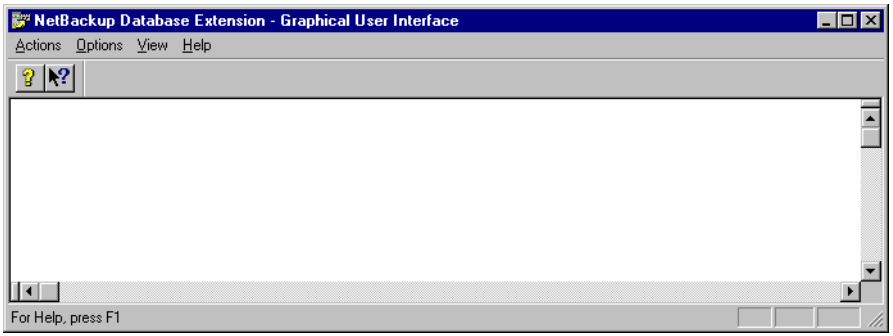
Sometimes SQL Server can skip the pre-write step if the restore statement is scripted with the `RESTORE` option. To see if this works, add the keywords "`RESTOREOPTION REPLACE`" to your restore batch file. If the replace option is used, then SQL Server does not check whether the database files are already in use before starting the restore.

Reference 12

This chapter provides reference information for the NetBackup for SQL Server, including:

- NetBackup for Microsoft SQL Server Graphical User Interface
- Menu Bar
- Dialogs

NetBackup for Microsoft SQL Server Graphical User Interface

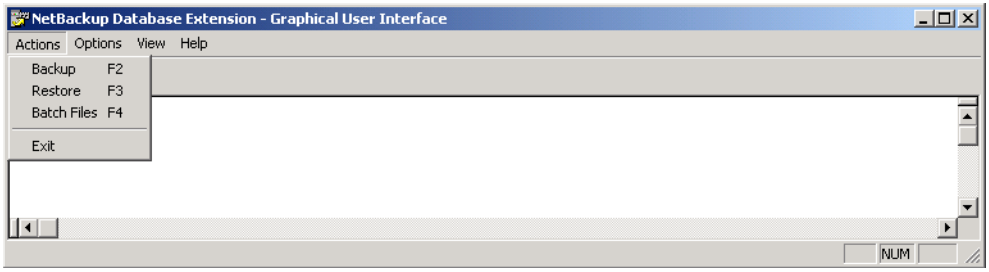


This window contains some familiar Windows elements: the Menu Bar, the Toolbar and the Status Bar. Refer to "Menu Bar" on page 211 and "View Menu" on page 213 for more information on these items.

Menu Bar

The menus on the Menu Bar have been developed for NetBackup for SQL Server.

Actions Menu



Use this menu to start a backup, restore or batch file operation.

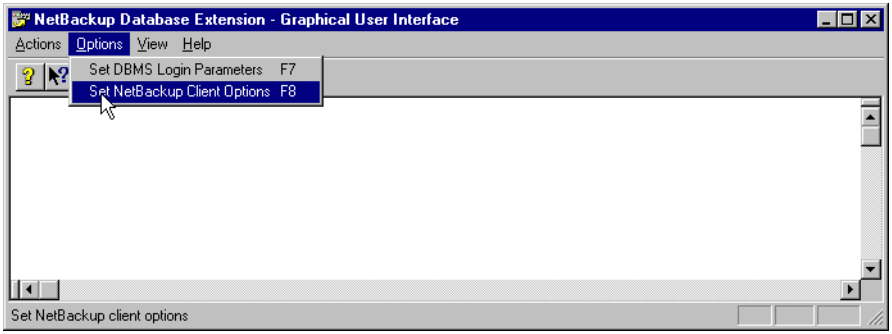
Backup This selection opens the Backup Microsoft SQL Server Objects dialog that allows you to perform a database backup, a database incremental backup, a database transaction log backup, a database filegroup backup and a database file backup. Refer to "Backup Microsoft SQL Server Objects" on page 217 for more details.

Restore This selection opens the Restore Microsoft SQL Server Objects dialog that allows you to perform a database restore, a database incremental restore, a database transaction log restore, a database filegroup restore and a database file restore. Refer to "Restore Microsoft SQL Server Objects" on page 224 for more details.

Batch Files Opens the Start Batch File dialog. Use this dialog to start batch files for NetBackup operations. Refer to "Start Batch File" on page 230 for more details.

Exit Use this command to exit the NetBackup for Microsoft SQL Server GUI.

Options Menu



Use this menu to set database management system (DBMS) options for your current session or for your defaults. You can also set certain NetBackup client options.

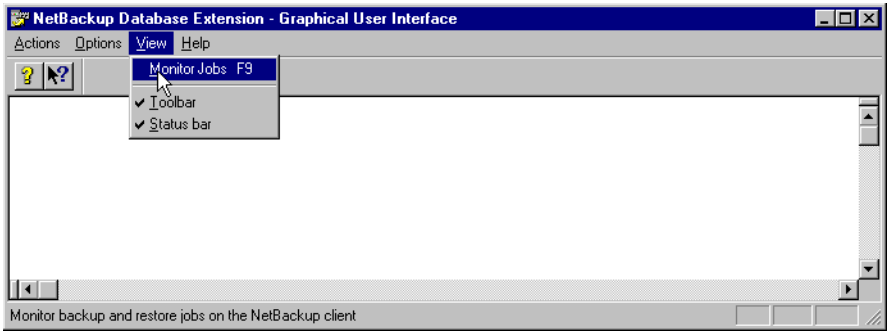
Set DBMS Login Parameters

Opens the Set database login parameters dialog. Use this dialog to select the SQL host and instance and to set the userid and password that you want to use for logging into SQL Server.

Set NetBackup Client Options

Opens the Client Options dialog. Use this dialog to set the current NetBackup for SQL Server trace level, and the block and certain tuning parameters.

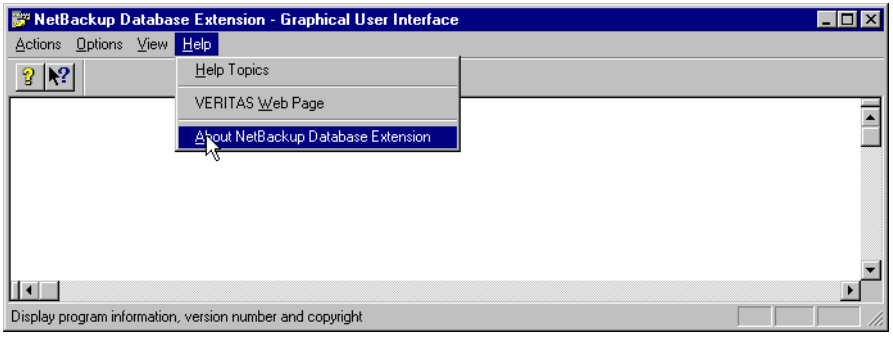
View Menu



Use this menu to view the status of NetBackup operations and to control the appearance of the Toolbar and the Status Bar.

Monitor Jobs	Opens the View Status dialog. Use this dialog to view the status of NetBackup operations.
Toolbar	Turns the Toolbar on or off.
Status Bar	Turns the Status Bar on or off.

Help Menu

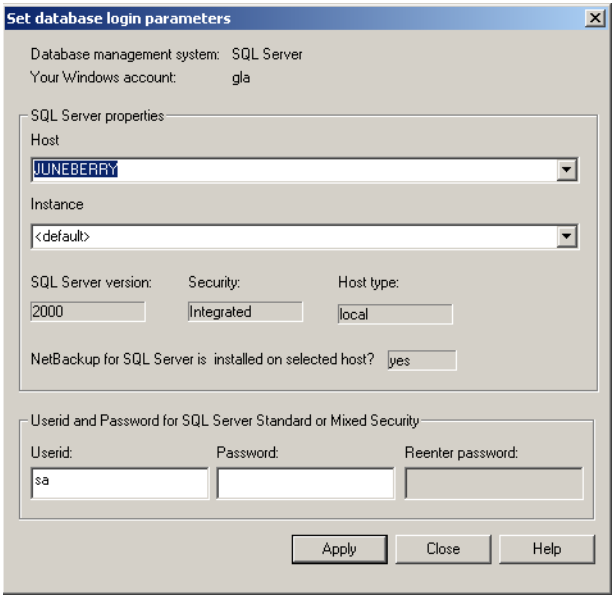


Use this menu to access the following help pages.

Help Topics	Opens the NetBackup Help topics. NetBackup Help contains the same information found in the user guide. You can also access NetBackup Help by using the context-sensitive button on the Toolbar. Refer to "NetBackup for Microsoft SQL Server Graphical User Interface" on page 146 for more details.
VERITAS Web Page	Displays the VERITAS world wide web page on your computer's default browser.
About NetBackup for SQL Server	Displays the version number of NetBackup for SQL Server.

Dialogs

Set database login parameters



To access this dialog choose **Options** > **Set DBMS login parameters**. Use this dialog to set the login parameters that NetBackup for SQL Server uses when logging into SQL Server. Refer to "Setting the Database Login Parameters" on page 63 for information about how to use this dialog.

SQL Server properties

Host — This item displays the current SQL Server host selection. Click on the drop-down list box to display all of the SQL Server hosts in the network. You can select a different SQL Server host from the drop-down list or type in the name directly.

Instance — This item displays the current SQL Server instance selection. Click on the drop-down list box to display all of the SQL Server instances on the selected host. You can select a different SQL Server instance from the drop-down list or you can type in the name directly.

Note The keyword <default> shown in the Instance box designates the default SQL Server.

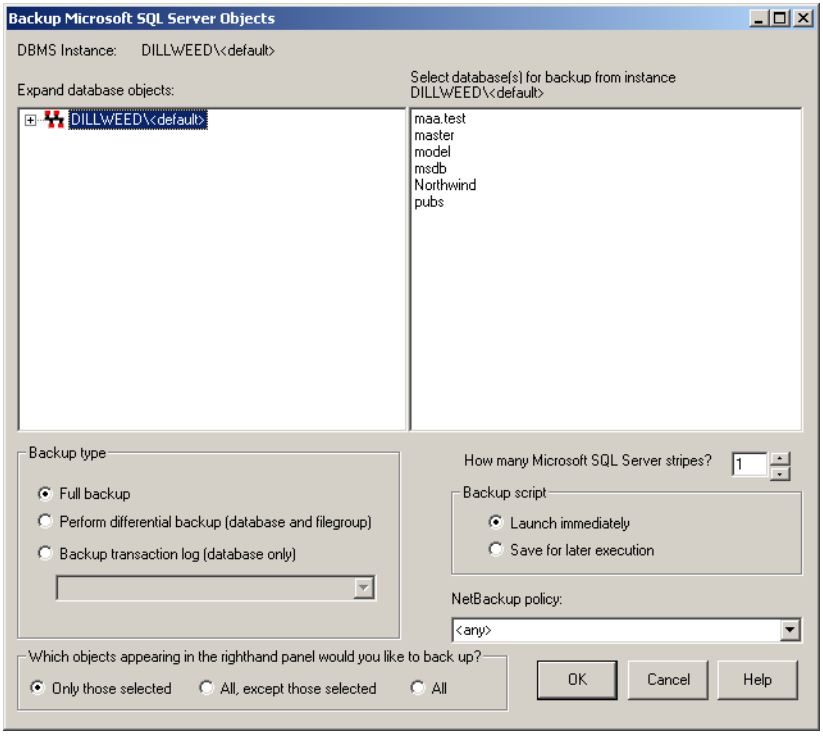
SQL Server version	Displays the version of the SQL instance selected in the **Host** and **Instance** boxes.
Security version	Displays the type of security being used by the SQL instance selected in the **Host** and **Instance** boxes.
Host type	Displays the host type local, remote, virtual or unknown.

Note The three previous items are updated when you click the **Apply** button.

Userid and Password for SQL Server Standard or Mixed Security

Userid	This box displays the username that NetBackup for SQL Server uses to log into SQL Server for your Windows user id.
Password, Reenter password	Use these fields to enter the SQL Server password that is associated with the Userid that you entered in the previous field.
Apply	To accept changes to the parameters, click this button.
Close	To return to the previous display click this button. If the **Apply** button has not been clicked, the changes to the parameters in this dialog are not retained.
Help	To view online help for this dialog, click this button.

Backup Microsoft SQL Server Objects



Access this dialog by choosing **Actions** > **Backup**. Use this dialog to initiate a backup operation.

Expand databases objects	This pane allows you to traverse live databases. You can expand the SQL Server instance to view its databases. Expanding each database will allow you to view its filegroups and you can expand a filegroup to view its files. You can select any object in this pane in order to view its constituent objects in the right-hand pane.
Select database(s) for backup from instance host\instance	This pane displays the list of constituent database objects of the highlighted host\instance in the left-hand pane. You can select one or more objects (databases) in this pane. (Use Ctrl + click and Shift + click to select multiple objects.)

Note The right pane changes in order to display the contents of the object displayed in the left pane. When an SQL instance is selected in the left pane, the right pane will display databases. When a database is selected in the left pane, the right pane displays filegroups. When a filegroup is selected in the left pane, the right pane displays files. Always use the right pane to select the objects that you want to back up.

Backup type

Full backup Select this option if you want to create a full database backup.

Perform differential backup (database and file group) Select this option if you want to create a differential backup image instead of a full backup.

Backup transaction log (database only) Select this option if you want to create a transaction log backup image. You can select one of the following transaction log options.

- **Back up and truncate transaction log** Back up the transaction log and remove the inactive part of the transaction log.
- **Back up transaction log but don't truncate it** Back up a transaction log without truncating it.
- **Truncate transaction log but don't back it up** Remove the inactive portion of the transaction log without backing it up.

How Many MS SQL Server Stripes? Use this box to set the number of backup stripes that you want SQL Server to create for your backup. Type a number from 1 to 32. Refer to "Configuring for Multi-Stream Operations" on page 45.

Which objects appearing in the righthand panel would you like to backup? Use this radio group to decide whether to backup:

- only the objects selected in the right-hand pane
- all of the objects in the right-hand pane except those selected
- all of the objects in the right-hand pane

Backup Script You can change these settings for current NetBackup operations.

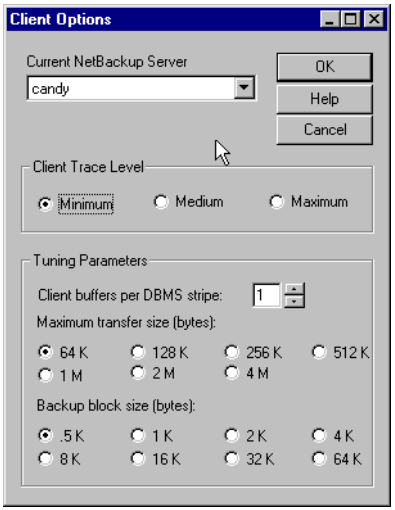
- **Launch Immediately** Select **Launch Immediately** if you wish to start a backup operation.
- **Save for Later Execution** Select **Save for Later Execution** to generate a script that can be started at a later time.

NetBackup Policy When the policy type is MS-SQL-Server and the host is a NetBackup master server or a remote console, then a drop-down list is available from which you can choose from the list of the policies defined on the server. Otherwise, type the NetBackup policy you want to use for this backup operation. The default is <any>.

If you keep this default, NetBackup will back up the database to the first active policy for your DBMS Server. You can select a different policy by typing its name into the box.

OK	To start a database backup or to generate a database backup script, select one or more databases and click this button. This button is enabled when you select an object in the right-hand pane.
Help	To view online help for this dialog, click this button.
Cancel	To cancel and return to the previous display without accepting changes to the parameters in this dialog, click this button.

Client Options



To access this dialog, choose **Options** > **Set NetBackup Client Options**. Use this dialog to set the following options for your current session.

Current NetBackup Server — Use this drop down menu to select the NetBackup Server that you wish to use for NetBackup For SQL Server Operations.

Note The NetBackup Server that you select will be associated with any NetBackup for SQL Server operations that are started under your current user account.

Client Trace Level Group — Use this radio group to set the trace level written to the `dbclient` log. Normally, you should set the level to Minimum. Maximum generates a very large amount of debug data.

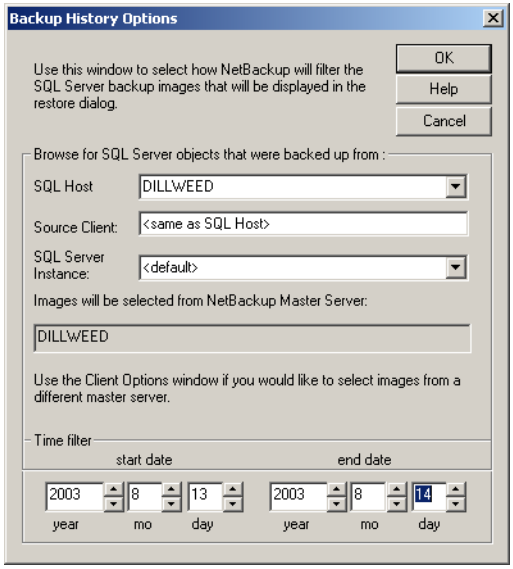
Tuning Parameters — The attributes in this box allow you to set the number of client buffers per DBMS stripe, the maximum transfer size in bytes, and the backup block size in bytes.

Client Buffers per DBMS Stripe — Use this parameter to set the number of buffers that SQL Server uses per database stripe.

Maximum Transfer Size (Bytes) — Use this radio group to set the maximum transfer size that SQL Server uses for backing up and restoring database objects. It can be set to any of the indicated values between 64 kilobytes and 4 Megabytes. Disabled for SQL Server 6.5.

Backup Block Size (Bytes)	Use this radio group to set the backup block size that SQL Server uses for backing up database objects. It can be set to any of the indicated values between 0.5 kilobytes and 64 kilobytes.
OK	Click this button to change your client options.
Help	To view online help for this dialog, click this button.
Cancel	To cancel and return to the previous display without accepting changes to the parameters in this dialog, click this button.

Backup History Options



This dialog is displayed when you choose **Actions** > **Restore**.

Use this dialog to:

- Set a different client on the NetBackup Server from which to browse for backup images
- Change the start and end times for which you would like to browse for images
- Set the SQL Server instance for which you would like to browse backup images.

SQL Host	From this list, choose the SQL Server host that you would like to generate a backup history for.
Source Client	Specifies the NetBackup client name for the selected client. This is the name used in the policy to identify the client.

Note If you use a specific network interface for backup, then the network interface name should be entered in the **Source Client** box. The network interface name is defined in the Host Properties for the server (open the properties for the server and click **Universal Settings**).

Note If the NetBackup master server is a UNIX machine, then you must provide the SQL hostname in the same case as it exists in the server's client list. For example, if the SQL host is "DILLWEED" but if it appears in the client list as "dillweed", then you should enter "dillweed" in to the **Source Client** field.

SQL Server Instance

From this list, choose the SQL Server instance that you would like to generate a backup history for.

Note The instance list in the Backup History Options dialog always contains a predefined entry that you should select if you wish to generate a history of backup images generated from a previous version of NetBackup for SQL Server. Choose NB_LEGACY_SQL7, if SQL Server 7.0 and/or 2000 are active and you wish to view 7.0 backup images that were generated by an earlier release of NetBackup. Choose NB_LEGACY_SQL65, if 6.5 is active and you wish to view 6.5 images generated from a previous version.

The attributes in this box will allow you to perform a redirected restore to a different client. Ask the NetBackup administrator to give you the permissions to perform a redirected restore to a different client.

Time Filter

Define the range of backups you want to search in this box.

Start Date

Enter the earliest date of the backup you want to search.

End Date

Enter the latest date of the backup you want to search.

OK

Click this button to change your backup history. After you click **OK** the Restore Microsoft SQL Server Objects dialog is displayed.

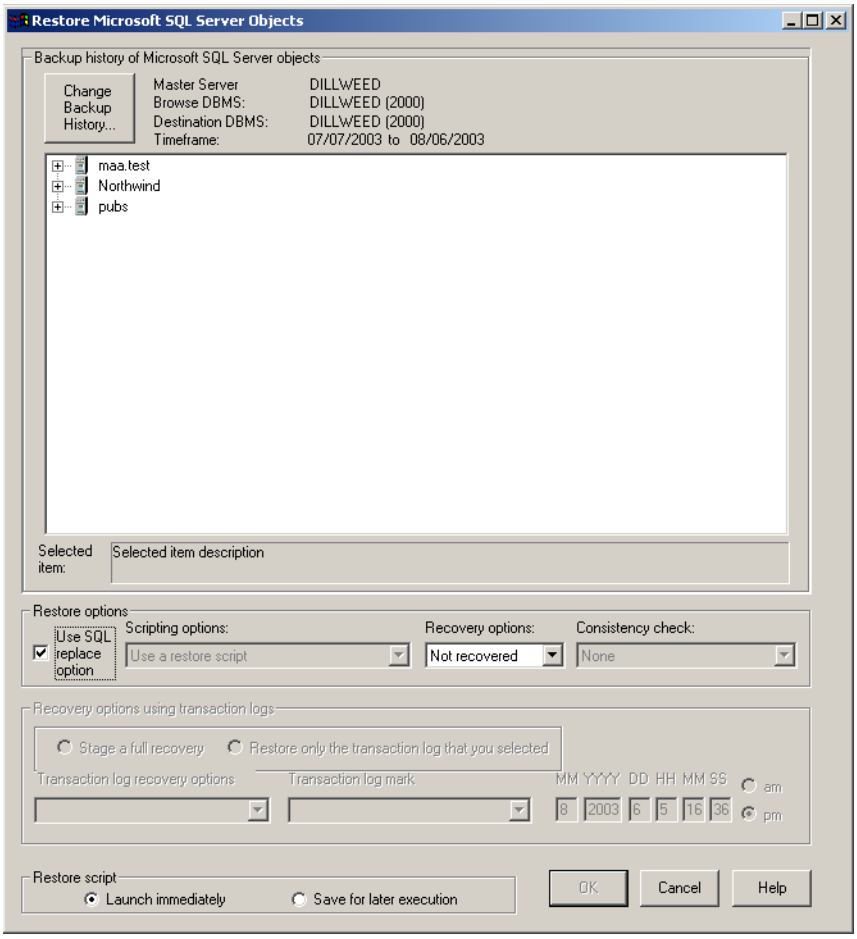
Help

To view online help for this dialog, click this button.

Cancel

To cancel and return to the previous display without accepting changes to the parameters in this dialog, click this button.

Restore Microsoft SQL Server Objects



To open this dialog, choose **Actions** > **Restore**, and after the Backup History Options dialog is displayed, click **OK**. Use this dialog to restore a database from a single object or a series of objects that constitute a full recovery set.

Backup history of Microsoft SQL Server Objects	This is a tree display of the database backup history for a NetBackup client. The backup client is typically the local host on which you are running NetBackup for SQL Server. However, it may be different if a different client has been specified through the Backup History Options dialog.
Change Backup History	Use this button to refresh the backup history window.

Note The icons used in the following expansions are depicted to the right of the Backup History pane.

You can expand and collapse portions of the tree by clicking on the **+** and **-** buttons. The tree can be expanded as follows:

Level 0: Database icons	Items at this level cannot be selected.
Level 1: Database backup images	The database backup images appearing at level 1 are full database backup images. Each is displayed with the date and time of the backup.
Level 1: Transaction log backup images	The transaction log backup images appear at level 1. Each is displayed with the date and time of the backup.
Level 1: Filegroup icons	Filegroup icons appear at level 1. These icons cannot be selected for restore.
Level 2: Database differential images	The database differential backup images appear at level 2 after expanding a database backup image with a + sign preceding the database icon. The database differential backup images are displayed with the date and time of the backup.
Level 2: Filegroup backup images	The filegroup backup images appear at this level when you expand a filegroup icon by clicking on the + sign preceding the filegroup icon. Each of these images is displayed with the date and time of the backup.
Level 2: File icons	The individual file icons appear at level 2. These icons cannot be selected for restore.
Level 3: Filegroup Differential images	The filegroup differential backup images appear at level 3 after expanding a filegroup backup image with a + sign preceding the filegroup backup. The filegroup differential backup images are displayed with the date and time of the backup.
Level 3: File images	The individual file backup images appear at this level when you expand a file icon by clicking on the + sign preceding the file icon. Each of these images is displayed with the date and time of the backup.

Note If you select a transaction log icon and the **Restore only the transaction log that you selected** box is not checked, then the following will automatically be selected: the parent Database backup icon, the closest preceding Database differential backup (if one exists), and all of the preceding transaction log icons. If you select a Database differential backup, then the parent Database backup icon is selected. The checkmarks indicate that these objects have been selected for staging in a full database restore.

Selected item — This text provides the name of the policy used to backup the selected object and the method of backup. These methods may be:

- Standard - conventional backup to tape or disk
- Frozen image
- Persistent frozen image
- Persistent frozen image to tape
- Remote frozen image

Restore Options — These options affect the restore operation.

Use SQL replace option — Select this option to restore with the SQL Server replace option.

Scripting options — Scripting options available when restoring from a database image.

Use a restore script — This option is the default and produces a script that performs a database restore.

Create a move template — Select this option if you want to create a script template for moving the selected database.

Create partial database restore template — Select this option if you want to create a script template for a partial database restore. This option is available only for SQL Server 2000.

Recovery options — This group contains the controls for you to specify one of the SQL Server recovery options.

Not recovered — Use this option during a restore if additional backup images must be applied to the database following the current restore. When you use this option, the database is left in a loading state.

Recovered — Select this option when restoring the last image in a restore sequence. After the recovery operation, the database is ready for use. If recover database is not performed, the database is in an intermediate state and is not usable. If Recovered is selected when an intermediate backup is being applied, you cannot continue to restore backups. You must restart the restore operation from the beginning.

Standby — Use this option during transaction log and database restore to create and maintain a standby database. Usage of this option requires a standby undo log, which by default, is placed in *install_path*\Logs\SQLStandBy\ When you use this option, the database is placed in 'standby' state following the restore.

Consistency Check — Select the consistency check to be performed after the restore. Output from the consistency check is written to the `dbclient` log. You cannot select consistency checking unless the database is being restored to the recovered state. If you choose consistency checking for a staged recovery, then the check occurs following the last restore.

None — Use this option if you do not want SQL Server to perform consistency checking.

Full check, excluding indexes — Select this to exclude indexes from the consistency check. If indexes are not checked, the consistency check runs significantly faster but is not as thorough. Only the data pages and clustered index pages for each user table are included in the consistency check. The consistency of the non-clustered index pages are not checked.

Full check, including indexes — Select this to include indexes in the consistency check. Any errors are logged.

Check catalog — Select this option to check for consistency in and between system tables in the specified database.

Physical check only — (SQL 2000) Select this to perform a low overhead check of the physical consistency of the SQL Server 2000 database. This option only checks the integrity of the physical structure of the page and record headers, and the consistency between the pages' object ID and index ID and the allocation structures.

Recovery options using transaction logs

Stage a full recovery — This option is enabled when a transaction log is selected.

Restore only the transaction log that you selected — This option is enabled when a transaction log is selected. Select this option to restore only the highlighted transaction log.

Transaction log recovery options

This group contains the controls for you to restore a transaction log to a point in time that precedes the time at which the transaction log was dumped. The individual entries in this group are only enabled if you have highlighted a transaction log backup.

If SQL Server 7.0 is active, then the only available option from this group is **To point in time**. The remaining options, which allow you to restore to a transaction mark, or to a time before or after a transaction mark, are enabled if you are working with an SQL Server 2000 database.

To point in time

Select this radio item to have the transaction log recovered to a point in time.

To transaction log mark

Select this radio item to have the transaction log recovered to a transaction log mark. With this option, you must enter a transaction log mark name.

To transaction log mark but after

Select this radio item to choose to have the transaction log recovered after to a transaction log mark but after a point in time. With this option, you must enter a transaction log mark name.

Before transaction log mark

Select this radio item if you would like to have the transaction log recovered to a point before the occurrence of a transaction log mark. With this option, you must enter a transaction log mark name.

Before transaction log mark but after

Select this radio item if you would like to have the transaction log recovered to a point before the occurrence of a transaction log mark but after a point in time. With this option, you must enter a transaction log mark name.

Transaction log mark

This pull-down menu is enabled if you have highlighted a database transaction log for restore, and the transaction log contains one or more transaction log marks, and you have selected one of the following radio buttons:

- To transaction log mark
- To transaction log mark but after
- Before transaction log mark
- Before transaction log mark but after.

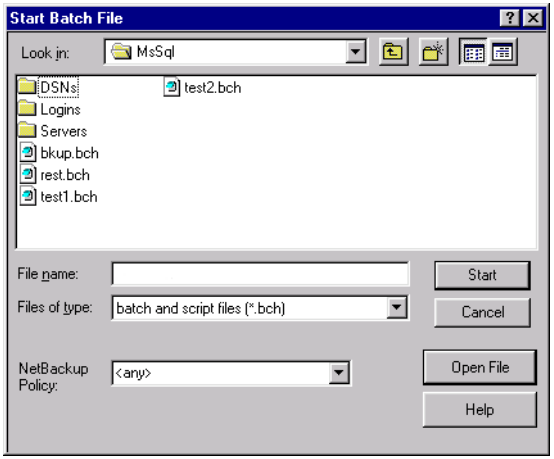
MM, YYYY, DD, HH, MM, SS am, pm

Use these fields to specify the time to which you want the transaction logs restored. Use these fields if you have selected one of the following radio buttons:

- To point in time
- To transaction log mark but after
- Before transaction log mark but after

Restore Script	You can change these settings for current NetBackup operations.
Launch Immediately	Select **Launch Immediately** if you wish to start a restore operation.
Save for Later Execution	Select **Save for Later Execution** to generate a script that can be started at a later time.
OK	To start a restore or to generate a restore script, select one or more objects in the backup history pane and click this button. This button is disabled if you have not selected any objects in the backup history pane.
Cancel	To cancel and return to the previous display without accepting changes to the parameters in this dialog, click this button.
Help	To view online help for this dialog, click this button.

Start Batch File



To access this dialog choose **Actions** > **Batch Files**. Use this dialog to start a NetBackup for SQL Server operation from a NetBackup for SQL Server batch file that you have written.

This dialog displays the file icons for the contents of the *install_path*\NetBackup\dbext\Mssql folder. You are generally advised to keep your NetBackup for SQL Server batch files there; however, support is provided through common Windows controls to navigate the file system to locate them anywhere.

Beside using this dialog to start operations, you can also use it to open your batch files in NotePad for viewing or editing.

General — This dialog uses the common controls from the Windows File menu. You can get additional help on many of the items in this dialog by pressing F1 and clicking on that item.

File Name — Type the name of the file you want to use for starting a batch NetBackup for SQL Server operation or to open the file in NotePad. You can also select the file icon.

File Type — By modifying the file extension, which appears in this box, you can control what files will be displayed in the main window.

NetBackup Policy	When the policy type is MS-SQL-Server and the host is a NetBackup master server or a remote console, then a drop-down list is available from which you can choose from the list of the policies defined on the server. Otherwise, type the name of the NetBackup policy on the master server that you want to back up to. If you use the default (<any>), NetBackup will back up to the first available policy. Note that the NetBackup policy is applicable only for backup operations, not for restores.
Start	Click this button to launch the NetBackup for SQL Server operation.
Help	To view online help for this dialog, click this button.
Cancel	To cancel and return to the previous display without accepting changes to the parameters in this dialog, click this button.
Open batch file	Click this button to open the selected file in NotePad.

Note See "Overview of Batch Files" on page 101 for a description of how to create a batch file.

Index

A

accessibility features xvi
activity logs *See* debug logs.
Activity Monitor 42, 43
alternate buffer method 57
alternate client 175
Application Backup schedule 19, 33
  backup window 19, 33
  configuration 19, 33
  retention 19, 33
authorization, for scheduled operations 47
automatic backup schedule 20, 33

B

Backup Exec
  importing images 179
  redirected restores 199
  restores
    client, specifying 186
    databases 187
    filegroups 193
    multiple databases 180
    options 181
    policy type, specifying 186
    server, specifying 186
    SQL Master Database 197
    tables 180
    transaction logs 189, 191
  supported SQL versions 180
Backup Selections list 23, 37
backups
  browsing for 75
  database filegroups 69
  database files 71
  databases 65, 111
  excluding databases 111
  image names, colon and backslash in 125
  in a Microsoft Cluster Server 137
  in a SAP environment 153, 154
    policy configuration 154
  in a SAP environment, scripts used 154
  in a VERITAS Cluster Server 137
  manual 42
  offhost 175
  preparing files and filegroups for 50
  SQL Server login 46
  SQL Server security restrictions 47
  tables 69, 71
  transaction logs 66
batch files
  BUFFER keyword 103
  default directory 10, 230
  defined in Backup Selections list 23, 37
  for backup operations 108
  running 120
  samples 10, 112–119
  specifying object to back up 105
  STRIPES keyword 107
  when required 101
BLOCKSIZE 55
bphdb log 205
bplist
  client parameter 125
  example 125
  server parameter 125
  SQL Server instance 125
  stripenumber 126
  total stripes 126
  using to retrieve list of backups 125
buffer space parameters
  BLOCKSIZE 55
  MAXTRANSFERSIZE 55

buffers
alternate buffer method 57
and transfer of data to NetBackup server 56

C

Clients list 25, 39
configuration
database debug level 206
Media Manager 13, 27
of NetBackup for MS-SQL, overview 9
of SQL Server 46
UNIX 26
Windows 12
copying databases *See* move operations.

D

database login parameters, setting 63
dbbackex 101
and client-based schedulers 61
DBMS (Database Management System) privileges 46
debug logs 205
disaster recovery
of databases 130
overview 128
rebuilding the master database 130
starting SQL Server in single-user mode 130

E

excluding databases from backups 111

G

Glossary. *See* NetBackup Help.

I

Inline Tape Copy 22, 36
installation 7
on Windows
in a cluster 134
prerequisites 7
verifying installed components 10
instance, SQL Server
displayed with bplist command 125
specified for restores 75
integrated security 46

J

jnbSA 26

L

license keys, registering 7
log shipping 54
login parameters, database, setting 63
login, SQL Server, for scheduled operations 46
logs, debug 205

M

manual backups
in a SAP environment 153
Maximum jobs per client 14, 28
MAXTRANSFERSIZE 55
Media Manager
configuration 13, 27
Microsoft Cluster Server (MSCS)
configuring the NetBackup Server 136
SQL Server backups in 137
SQL Server restores in 137
using with NetBackup for SQL Server 134
VIRTUALSERVER setting 134
MOVE keyword 104
move operations
sample template 87
multi-interface network connections
and clusters 133-144
backups 141–143, 149
configuration 138–141, 146-148
overview138, 145
restores 143–144, 150–151
multiple data streams *See* striping
multiplexing
and multiple data streams
and performance 56

N

NetBackup Administration Console, launching 12, 26
NetBackup Client Service 47
NetBackup for Microsoft SQL Server, starting 62
NetBackup Server Scheduler 101
network interface 222
NUMBER OF BUFFERS PER STRIPE 56

O

ODBC, data source names, in batch files 104
offhost backups 175

P

partial database restores 89
policy configuration
  adding new 16, 29
  attributes 18, 32
  Backup Selections list 23, 37
  clients 25, 39
  overview 15, 29
  schedules 19
  testing 41, 42
preventing timeout failures 208
progress reports 208

R

R/3 database, restoring 156
recovery factors 51
  recovery strategies 51
  transaction logs 51
redirected restores
  of databases, to different locations 96
  to different hosts 95
remote hosts, browsing 122
reports
  bphdb 205
  dbclient 205
  progress 208
restores
  and striping 124
  browsing for backup images 75
  database differentials 77
  database files 80
  database move *See* move operations.
  databases 76
  filegroup differentials 79
  filegroups 78
  in a Microsoft Cluster Server 137
  in a SAP environment 153
  in a VERITAS Cluster Server 137
  partial databases 90
  R/3 database 156
  redirecting *See* redirected restores.
  specifying SQL Server instance 75
  SQL Server login 46
  transaction logs
    within a staged recovery 84
    without staging a full recovery 82
retention period 22, 36

S

SAP environment
  backups 153, 154
  backups, policy configuration 154
  backups, scripts used 154
  manual backups 153
  restores 153
  transaction log backups 155
schedulers, client-based, and dbbackex 61
schedules
  adding 19, 32
  frequency 21, 35
  properties 21, 35
  retention 22, 36
security
  integrated 46
  SQL Server restrictions 47
  standard SQL Server security 46, 49
shared memory, and performance 56
Source Client 75
  specifying a network interface 222
SQL Host 75
SQL Server Instance 75
SQL Server login, for scheduled operations 46
SQL Server, configuration 46
standard security 46
standby server 54
STRIPES keyword 107
striping
  and automatic restore 124
  and multiplexing
  and number of buffers 103
  and performance 56
  configuring for non-multiplexed 45
  invoking using the STRIPE parameter 124
  maximum jobs per client 14, 28

T

testing policy configuration 41, 42
timeout failures, preventing 208

U

Use alternate client 175

V

VERITAS Cluster Server (VCS)
configuring the NetBackup Server 136
SQL Server backups in 137
SQL Server restores in 137
VIRTUALSERVER 134